

CFM 2409 VS



日本特許庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出願年月日

Date of Application:

2000年12月27日

出願番号

Application Number:

特願2000-399331

出願人

Applicant(s):

キヤノン株式会社

RECEIVED

MAR 14 2002

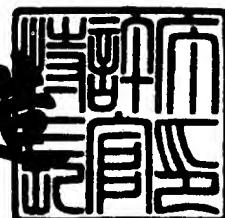
Technology Center 2100

CERTIFIED COPY OF
PRIORITY DOCUMENT

2001年11月16日

特許長官
Commissioner,
Japan Patent Office

及川耕造



出証番号 出証特2001-3100529

【書類名】 特許願
【整理番号】 4367014
【提出日】 平成12年12月27日
【あて先】 特許庁長官殿
【国際特許分類】 G06F 1/00
【発明の名称】 データ処理装置及びその制御方法、データ処理方法、コンピュータ可読メモリ
【請求項の数】 11
【発明者】
【住所又は居所】 東京都大田区下丸子3丁目30番2号 キヤノン株式会社内
【氏名】 中山 忠義
【特許出願人】
【識別番号】 000001007
【氏名又は名称】 キヤノン株式会社
【代理人】
【識別番号】 100076428
【弁理士】
【氏名又は名称】 大塚 康徳
【電話番号】 03-5276-3241
【選任した代理人】
【識別番号】 100101306
【弁理士】
【氏名又は名称】 丸山 幸雄
【電話番号】 03-5276-3241
【選任した代理人】
【識別番号】 100115071
【弁理士】
【氏名又は名称】 大塚 康弘

【電話番号】 03-5276-3241

【手数料の表示】

【予納台帳番号】 003458

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0001010

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 データ処理装置及びその制御方法、データ処理方法、コンピュータ可読メモリ

【特許請求の範囲】

【請求項1】 データを処理するデータ処理装置であって、データを保持する保持手段と、複数の加算器もしくは減算器もしくは加減算器のいずれかと、更に乗算器もしくは位取り変換器とを有する演算ユニットとを複数個備え、

前記複数の演算ユニットは、 n 個従属に接続した構成からなり、入力する $2^n + 1$ タップと $2^n - 1$ タップのデータから、2種類のウェーブレット変換係数の演算を行う

ことを特徴とするデータ処理装置。

【請求項2】 前記演算ユニットは、前記乗算器もしくは位取り変換器の出力を、前記保持手段の入力もしくは出力データのどちらか一方、あるいは入力・出力データの両方に加算もしくは減算できるように、該保持手段の前後に2つの加算器もしくは減算器もしくは加減算器を有する

ことを特徴とする請求項1に記載のデータ処理装置。

【請求項3】 前記演算ユニットは、前記位取り変換器を2つ有し、前記演算に応じて2種類の位取りを切り換える

ことを特徴とする請求項2に記載のデータ処理装置。

【請求項4】 前記演算ユニットは、該演算ユニットへの入力データの1つに前記保持手段の出力データを加算すると共に、前記位取り変換器を複数有し、前記演算に応じて2種類の位取りを切り換える

ことを特徴とする請求項1に記載のデータ処理装置。

【請求項5】 前記演算ユニットは、前記加算器もしくは減算器にオフセットを入力するオフセット生成手段と、該演算ユニット間に配置された該演算ユニットの出力データの丸め処理を行う丸め処理手段を有する

ことを特徴とする請求項3あるいは請求項4に記載のデータ処理装置。

【請求項6】 前記オフセット生成手段と丸め処理手段それぞれにおいて、

データをマスクする第1マスク手段と第2マスク手段を有し、第1マスク手段と第2マスク手段とが排他的に有効となるように制御する

ことを特徴とする請求項5に記載のデータ処理装置。

【請求項7】 前記演算ユニットは、前記保持手段を少なくとも2つ有し、前記2つの保持手段それぞれに2ラインのデータをライン交互に入力して、前記演算を行う

ことを特徴とする請求項1に記載のデータ処理装置。

【請求項8】 データを処理するデータ処理装置の制御方法であって、データを保持する保持手段と、複数の加算器もしくは減算器もしくは加減算器のいずれかと、更に乗算器もしくは位取り変換器とを有する演算ユニットとがn個從属に接続されて構成された演算ユニット群の先頭の演算ユニットへ、データを入力する入力工程と、

入力した $2^n + 1$ タップと $2^n - 1$ タップのデータから、ウェーブレット変換係数の演算を行うために、前記演算ユニット群を制御する制御工程とを備えることを特徴とするデータ処理装置の制御方法。

【請求項9】 データを処理するデータ処理方法であって、データを入力する入力工程と、前記入力工程で入力されたデータに基づいて、Lifting格子構造上の格子点データを演算する途中のデータである中間データを生成する第1生成工程と

前記生成工程で生成された中間データを記憶媒体に保持する保持工程と、前記保持工程で保持された中間データを用いて、前記格子点データを生成する第2生成工程と

を備えることを特徴とするデータ処理方法。

【請求項10】 データを処理するデータ処理装置の制御のプログラムコードが格納されたコンピュータ可読メモリであって、

データを保持する保持手段と、複数の加算器もしくは減算器もしくは加減算器のいずれかと、更に乗算器もしくは位取り変換器とを有する演算ユニットとがn個從属に接続されて構成された演算ユニット群の先頭の演算ユニットへ、データ

を入力する入力工程のプログラムコードと、

入力した $2^n + 1$ タップと $2^n - 1$ タップのデータから、ウェーブレット変換係数の演算を行うために、前記演算ユニット群を制御する制御工程のプログラムコードと

を備えることを特徴とするコンピュータ可読メモリ。

【請求項 11】 データを処理するデータ処理のプログラムコードが格納されたコンピュータ可読メモリであって、

データを入力する入力工程のプログラムコードと、

前記入力工程で入力されたデータに基づいて、Lifting 格子構造上の格子点データを演算する途中のデータである中間データを生成する第1生成工程のプログラムコードと、

前記生成工程で生成された中間データを記憶媒体に保持する保持工程のプログラムコードと、

前記保持工程で保持された中間データを用いて、前記格子点データを生成する第2生成工程のプログラムコードと

を備えることコンピュータ可読メモリ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、データを処理するデータ処理装置及びその制御方法、データ処理方法、コンピュータ可読メモリに関するものである。

【0002】

【従来の技術】

画像、特に、多値画像は非常に多くの情報を含んでおり、その画像を蓄積・伝送する際にはデータ量が膨大になってしまうという問題がある。このため画像の蓄積・伝送に際しては、画像の持つ冗長性を除く、あるいは画質の劣化が視覚的に認識し難い程度で画像データの変化を許容することによってデータ量を大幅に削減する高能率符号化が用いられる。

【0003】

例えば、静止画像の国際標準符号化方式としてISOとITU-Tにより勧告されたJPEGでは、画像データをブロック（8画素×8画素）ごとに離散コサイン変換（DCT）した後に、変換によって得られる各変換係数を各々量子化し、さらにエントロピー符号化することにより画像データを圧縮している。この圧縮では、画像データをブロックごとにDCT、量子化を行なっているので、復号画像の各ブロックの境界で、所謂ブロック歪みが見える場合がある。

【0004】

一方、新しい静止画像の国際標準符号化方式として検討されているJPEG2000では、量子化・エントロピー符号化の前に行なう変換処理として、ウェーブレット変換が用いられている。ウェーブレット変換は、DCT変換のように小さなブロック単位で処理を行うのではなく、該ブロックより十分大きなサイズのタイル単位で変換処理をするので、前記ブロック歪みがないといった特徴がある。

【0005】

以下では、Lifting Schemeを用いたウェーブレット変換フィルタ処理について説明する。

【0006】

JPEG2000で使われているウェーブレット変換は、Lifting Schemeという方法で処理をすると、少ない演算量で効率良く変換処理を行なうことができる。

【0007】

図1に順方向のLifting Scheme、図2に逆方向のLifting Schemeのシグナルフローを示す。図1、図2は、低域ウェーブレット変換係数の演算に9タップのデータ、高域ウェーブレット変換係数の演算に7タップのデータを用いる場合のシグナルフローである。図の中の α 、 β 、 γ 、 δ はLifting係数と呼ばれるものである。

【0008】

以下、図1の動作について説明する。

【0009】

入力画素を $X_0, X_1, X_2, X_3, X_4, X_5, \dots, X_n$ のように順に表わす。該入力画素は、分類ユニット 201 にて、偶数画素系列と奇数画素系列とに分類され、該分類ユニット 201 の上側には添字が偶数の画素 $X_0, X_2, X_4, \dots, X_{2n}$ が、下側には添字が奇数の画素 $X_1, X_3, X_5, \dots, X_{2n+1}$ が出力される。

初段の Lifting 处理では、偶数画素系列に対し Lifting 係数 : α を乗算し、連続する 2 個の乗算結果を、該 2 画素の中央に位置する奇数画素系列中の画素に加算する。

【0010】

これを一般化した式で表現すると、以下のようになる。

【0011】

$$D_{2n+1} = X_{2n+1} + \alpha \cdot X_{2n} + \alpha \cdot X_{2n+2} \quad (1)$$

2 段目の Lifting 处理では、新たに得られた奇数画素系列 $D_1, D_3, D_5, \dots, D_{2n+1}$ に対し Lifting 係数 : β を乗算し、連続する 2 個の乗算結果を、該 2 画素の中央に位置する偶数画素系列中の画素に加算する。

【0012】

これを一般化した式で表現すると、以下のようになる。

【0013】

$$E_{2n+2} = X_{2n+2} + \beta \cdot D_{2n+1} + \beta \cdot D_{2n+3} \quad (2)$$

3 段目の Lifting 处理では、Lifting 係数 : γ を用いて、初段と同じように、4 段目の Lifting 处理では、Lifting 係数 : δ を用いて、2 段目と同じように処理する。3 段、4 段目の Lifting 处理内容を一般化した式で表現すると、以下のようになる。

【0014】

$$H_{2n+1} = D_{2n+1} + \gamma \cdot E_{2n} + \gamma \cdot E_{2n+2} \quad (3)$$

$$L_{2n+2} = E_{2n+2} + \delta \cdot H_{2n+1} + \delta \cdot H_{2n+3} \quad (4)$$

図中、K は、低域・高域ウェーブレット係数を正規化するものであるが、本発明の本質を説明するにあたって、特に関係ないことであるので、以下、説明を省略する。

【0015】

正規化処理を無視すれば、3段、4段目のLifting Scheme処理によって得られる H_n , L_n は各々ウェーブレット高域変換係数とウェーブレット低域変換係数に対応する。

【0016】

次に、図2に示す逆方向のLifting Schemeのシグナルフローについて簡単に説明する。まず始めに、順方向のLifting Schemeにおける正規化処理に対応して、逆の係数を掛けた後、4段のLifting処理を行なう。各段の処理内容を一般化した式で表現すると、以下のようになる。

$$(1 \text{ 段目}) \quad E_{2n+2} = L_{2n+2} - \delta \cdot H_{2n+1} - \delta \cdot H_{2n+3} \quad (5)$$

$$(2 \text{ 段目}) \quad D_{2n+1} = H_{2n+1} - \gamma \cdot E_{2n} - \gamma \cdot E_{2n+2} \quad (6)$$

$$(3 \text{ 段目}) \quad X_{2n+2} = E_{2n+2} - \beta \cdot D_{2n+1} - \beta \cdot D_{2n+3} \quad (7)$$

$$(4 \text{ 段目}) \quad X_{2n+1} = D_{2n+1} - \alpha \cdot X_{2n} - \alpha \cdot X_{2n+2} \quad (8)$$

上記(5)～(8)式は、各々(4)～(1)式を移項して得られるものである。

【0017】

以上が、Lifting Schemeを用いたウェーブレット変換フィルタ処理についての説明であり、次に、該Lifting Schemeに再帰的演算を組み合わせたウェーブレット変換フィルタ処理について説明する。

【0018】

図1及び図2のLifting Schemeを別の観点から表現したものが、図3及び図4に示すLifting格子構造である。同図において、□は入力データを、○は格子点（あるいは格子点データ演算器）を表わし、○から出ている矢印は格子点データの流れを示す。これらの図はLifting Schemeにおける基本処理（前記(1)～(8)式の処理）並びに該処理によって得られる新たなデータを1つの格子点に対応させたものである。

【0019】

図3に示す順方向変換のLifting格子構造では、1つの格子点データは前記(1)～(4)式のいずれかを用いて計算される。

【0020】

図4に示す逆方向変換のLifting格子構造では、1つの格子点データは前記(5)～(8)式のいずれかにより計算される。

【0021】

Lifting格子構造を見ることによって、入力データや各格子点データの依存関係が一目瞭然となる。例えば、変換出力データであるL4を計算するには、9つの入力データ：X0～X8が必要であるが、3つの格子点データ：H3、E4、H5だけからも計算できることが分かる。

【0022】

L4、H5を計算し出力した時点で、X8、D7、E6、H5の4つのデータを残しておくことが可能である。この4つのデータと新たな入力データX9、X10とから、D9、E8、H7、L6の順に4つの格子点データを計算することができる。そして、L6とH7を変換データとして出力すると共に、X10、D9、E8、H7の4つのデータを残しておいて次の計算に利用するといった効率的な演算処理が可能である。

【0023】

ところで、この演算処理を効率的に行うには、先頭から偶数添え字の画素データと奇数添え字の画素データのペアを1つの単位にして処理する必要がある。上記の説明では、入力する画素データのペアが奇数添え字から始まるため、先頭の偶数添え字の画素データが半端になってしまふ。

【0024】

水平方向のウェーブレット変換処理の場合、半端となるのは先頭1画素だけであるが、垂直方向のウェーブレット変換処理では、先頭1ラインのデータが半端なデータとなってしまう。その上、大部分の画像データの垂直サイズは偶数なので、最後の1ラインも半端になってしまふ。

【0025】

ハードウェアで処理した場合、ペアの2ラインを処理する時間と先頭の1ラインのみを処理する時間は同じであるので、画像の先頭と最後で生ずる半端な1ラインは処理効率が悪い。

【0026】

【発明が解決しようとする課題】

上述したように、Left shifting処理の処理結果を保存しておき、それを再利用することによって効率的な演算処理をしようとするとき、データストリームの途中のデータは2ライン同時に処理ができるが、偶数サイズの画像では先頭と最後のラインのみ1ライン単独で処理しなければならず効率が良くない。

【0027】

本発明は上記の課題を解決するためになされたものであり、効率的にウェーブレット変換処理を実行できるデータ処理装置及びその制御方法、データ処理方法、コンピュータ可読メモリを提供することを目的とする。

【0028】

【課題を解決するための手段】

上記の目的を達成するための本発明によるデータ処理装置は以下の構成を備える。即ち、

データを処理するデータ処理装置であって、

データを保持する保持手段と、複数の加算器もしくは減算器もしくは加減算器のいずれかと、更に乗算器もしくは位取り変換器とを有する演算ユニットとを複数個備え、

前記複数の演算ユニットは、 n 個従属に接続した構成からなり、入力する $2^n + 1$ タップと $2^n - 1$ タップのデータから、2種類のウェーブレット変換係数の演算を行う。

【0029】

また、好ましくは、前記演算ユニットは、前記乗算器もしくは位取り変換器の出力を、前記保持手段の入力もしくは出力データのどちらか一方、あるいは入力・出力データの両方に加算もしくは減算できるように、該保持手段の前後に2つの加算器もしくは減算器もしくは加減算器を有する。

【0030】

また、好ましくは、前記演算ユニットは、前記位取り変換器を2つ有し、前記演算に応じて2種類の位取りを切り換える。

【0031】

また、好ましくは、前記演算ユニットは、該演算ユニットへの入力データの1つに前記保持手段の出力データを加算すると共に、前記位取り変換器を複数有し、前記演算に応じて2種類の位取りを切り換える。

【0032】

また、好ましくは、前記演算ユニットは、前記加算器もしくは減算器にオフセットを入力するオフセット生成手段と、該演算ユニット間に配置された該演算ユニットの出力データの丸め処理を行う丸め処理手段を有する。

【0033】

また、好ましくは、前記オフセット生成手段と丸め処理手段それぞれにおいて、データをマスクする第1マスク手段と第2マスク手段を有し、第1マスク手段と第2マスク手段とが排他的に有効となるように制御する。

【0034】

また、好ましくは、前記演算ユニットは、前記保持手段を少なくとも2つ有し、前記2つの保持手段それぞれに2ラインのデータをライン交互に入力して、前記演算を行う。

【0035】

上記の目的を達成するための本発明によるデータ処理装置の制御方法は以下の構成を備える。即ち、

データを処理するデータ処理装置の制御方法であって、

データを保持する保持手段と、複数の加算器もしくは減算器もしくは加減算器のいずれかと、更に乗算器もしくは位取り変換器とを有する演算ユニットとがn個従属に接続されて構成された演算ユニット群の先頭の演算ユニットへ、 $2^n + 1$ タップと $2^n - 1$ タップのデータを入力する入力工程と、

前記入力した $2^n + 1$ タップと $2^n - 1$ タップのデータによって、ウェーブレット変換係数の演算を行うために、前記演算ユニット群を制御する制御工程とを備える。

【0036】

上記の目的を達成するための本発明によるデータ処理方法は以下の構成を備える。即ち、

データを処理するデータ処理方法であって、

データを入力する入力工程と、

前記入力工程で入力されたデータに基づいて、Lifting格子構造上の格子点データを演算する途中のデータである中間データを生成する第1生成工程と

前記生成工程で生成された中間データを記憶媒体に保持する保持工程と、

前記保持工程で保持された中間データを用いて、前記格子点データを生成する第2生成工程と

を備える。

【0037】

上記の目的を達成するための本発明によるコンピュータ可読メモリは以下の構成を備える。即ち、

データを処理するデータ処理装置の制御のプログラムコードが格納されたコンピュータ可読メモリであって、

データを保持する保持手段と、複数の加算器もしくは減算器もしくは加減算器のいずれかと、更に乗算器もしくは位取り変換器とを有する演算ユニットとがn個従属に接続されて構成された演算ユニット群の先頭の演算ユニットへ、データを入力する入力工程のプログラムコードと、

前記入力した $2n+1$ タップと $2n-1$ タップのデータから、ウェーブレット変換係数の演算を行うために、前記演算ユニット群を制御する制御工程のプログラムコードと

を備えることを特徴とするコンピュータ可読メモリ。

【0038】

上記の目的を達成するための本発明によるコンピュータ可読メモリは以下の構成を備える。即ち、

データを処理するデータ処理のプログラムコードが格納されたコンピュータ可読メモリであって、

データを入力する入力工程のプログラムコードと、
 前記入力工程で入力されたデータに基づいて、Lifting格子構造上の格子点データを演算する途中のデータである中間データを生成する第1生成工程のプログラムコードと、
 前記生成工程で生成された中間データを記憶媒体に保持する保持工程のプログラムコードと、
 前記保持工程で保持された中間データを用いて、前記格子点データを生成する第2生成工程のプログラムコードと
 を備える。

【0039】

【発明の実施の形態】

以下、図面を参照して本発明の好適な実施形態を詳細に説明する。
 <実施形態1>

本発明では、従来例で述べた「L4、H5を計算し出力した時点でX8、D7、E6、H5の4つのデータを残しておき、新たなデータX9とX10を入力する」といった処理を次のように変える。

【0040】

すなわち、図3において「L4、H5を計算し出力した時点でD9、E8、H7、L6の4つのデータを演算する途中のデータ（以下、中間データとも称する）D9t、E8t、H7t、L6tを残しておき、次の処理サイクルで新たなデータX10とX11を入力する」といった処理を行う。

【0041】

尚、実施形態1では、低域・高域ウェーブレット変換係数の演算にそれぞれ、9タップと7タップのデータを用いる場合、つまり、 9×7 フィルタ（9タップと7タップのデータからなるフィルタ）の場合について説明するが、これに限定されず、本発明は、低域・高域ウェーブレット変換係数の演算にそれぞれ、 $2^n + 1$ タップと $2^n - 1$ タップのデータに対して適用できる。

【0042】

まず、上述の中間データは、以下のように示される。

【0043】

$$D_9t = X_9 + \alpha \cdot X_8 \quad (9)$$

$$E_8t = X_8 + \beta \cdot D_7 \quad (10)$$

$$H_7t = D_7 + \gamma \cdot E_6 \quad (11)$$

$$L_6t = E_6 + \delta \cdot H_5 \quad (12)$$

同時に、前の処理サイクルで残しておいた D_7t 、 E_6t 、 H_5t 、 L_4t から、 L_4 、 H_5 を以下の演算で計算する。

【0044】

$$D_7 = D_7t + \alpha \cdot X_8 \quad (13)$$

$$E_6 = E_6t + \beta \cdot D_7 \quad (14)$$

$$H_5 = H_5t + \gamma \cdot E_6 \quad (15)$$

$$L_4 = L_4t + \delta \cdot H_5 \quad (16)$$

残しておくデータはそれぞれレジスタで保持しておいて次の処理サイクルで利用する。該データを用いて次の処理サイクルで行なう演算は以下の内容である。

【0045】

$$D_9 = D_9t + \alpha \cdot X_{10} \quad (17)$$

$$E_8 = E_8t + \beta \cdot D_9 \quad (18)$$

$$H_7 = H_7t + \gamma \cdot E_8 \quad (19)$$

$$L_6 = L_6t + \delta \cdot H_7 \quad (20)$$

上記処理によって、 L_6 、 H_7 を出力することができる。

【0046】

この処理内容を *Lefting* 格子構造の図にならって表現したのが図5である。実際のハードウェア構成を図6に示す。同図において、

601、603は、上記ペアデータを入力する端子である。605、607は、それぞれ低域ウェーブレット変換係数と高域ウェーブレット変換係数を出力する端子である。611は、*Lefting* 係数を乗算係数として乗算する乗算器である。613は、演算途中のデータである中間データを保持するレジスタである。615、617は、乗算器の出力をレジスタの入力と出力データに加算する加算器である。621は、*Lefting* 演算ユニットである。622、623

、624は、Lifting演算ユニット621とは乗算器の乗算係数が違うだけで他はすべて同じ構成のLifting演算ユニットである。

【0047】

入力端子601と603に、X8とX9を入力している時、乗算器611にて、X8にLifting係数 α が乗算され、乗算結果として $\alpha \cdot X8$ が出力される。この乗算結果は、それぞれ加算器615、617に送られ、加算器615では前記(9)式、加算器617では、前記(13)式が演算される。

【0048】

他のLifting演算ユニット622、623、624においても下記に示す対応関係で演算処理が行なわれる。すなわち、

Lifting演算ユニット622では、前記(10)式と(14)式

Lifting演算ユニット623では、前記(11)式と(15)式

Lifting演算ユニット624では、前記(12)式と(16)式の演算が同時に行なわれる。

【0049】

よって、Lifting演算ユニット624では低域ウェーブレット変換係数L4が、Lifting演算ユニット623では高域ウェーブレット変換係数H5が演算され、それらの係数は端子605、607から出力される。(9)～(12)式の演算結果は、各演算ユニット内のレジスタに保持され、次のサイクルで実行される(17)～(20)式の演算に利用される。

【0050】

以上のようにして、水平方向の順方向のウェーブレット変換処理を行なうことができる。

【0051】

本発明人は、図7に示す格子点演算ユニットを多段接続してウェーブレット変換処理を行なうデータ処理装置について既に提案している。この提案内容と本発明の内容の違いは以下の2点である。

【0052】

(i) 格子点演算ユニットとLifting演算ユニットの内部構成の違い

(ii) 同時に処理する2つのデータの組み合わせの違い（入力位相の違いとも言える）

その他については基本的に同じである。特に、それらの演算ユニットを多段接続して処理する接続方法に関してはまったく同じであり、それらの演算ユニットをブラックボックス化して、入力データの位相を無視すれば、2つの処理系はまったく同じに見える。

【0053】

よって、前記格子点データ演算ユニットをLifting演算ユニットに置き換えるだけで、既に提案した内容における応用例は、本発明に対してそのまま適用できる。

【0054】

応用例1 水平方向の逆方向ウェーブレット変換処理装置

応用例2 垂直方向の順方向ウェーブレット変換処理装置

応用例3 垂直方向の逆方向ウェーブレット変換処理装置

応用例4 異なる種類のデータを多重化処理可能なウェーブレット変換処理装置

応用例5 順方向と逆方向の切換可能なウェーブレット変換処理装置（水平・垂直）

応用例6 上記応用例5において係数正規化用の乗算器を共用したウェーブレット変換処理装置

上記各応用例について簡単に説明する。

【0055】

応用例1は図8に示す構成で実現される。図6の構成との違いは、Lifting係数の順序が上から δ 、 γ 、 β 、 α となっており、図6に対して上下逆の順序になっている。また、図6におけるレジスタ前後の加算器は、図8ですべて減算器に置き換わっている。（係数として $-\delta$ 、 $-\gamma$ 、 $-\beta$ 、 $-\alpha$ を用いるのなら加算器のままでもよい。）

応用例2と3は、それぞれ図6と図8の各演算ユニット内のレジスタをすべてラインメモリに置き換えるだけでよい。（図面は省略する。）

応用例4は、図6の各演算ユニット内のレジスタを、1個から複数個の従属接続に変更し、複数種類のデータを多重化してから、端子601, 603へ入力することで実現できる。

【0056】

応用例5は、図6の各演算ユニット内の加算器をすべて加減算器に置き換え、順方向変換時には加算器として使用し、逆方向変換時には減算器として使用すると共に、演算ユニット間にセレクタを配して、処理したデータが下の演算ユニットからすぐ上の演算ユニットへ流れるように制御し、図8の構成と等価になるようにしたものである。該構成を図9に示す。

【0057】

応用例6は応用例5に対し、さらにもう1つセレクタを設け、係数正規化用の乗算器を順方向変換時と逆方向変換時で共用できるようにしたものである。該構成を図10に示す。

【0058】

7番目の応用例として、

応用例7 画像境界部のデータを拡張することなく境界処理が可能なウェーブレット変換処理装置
である。

【0059】

境界処理を一言で説明すると、「格子点データの演算に必要な3つのデータの内、2つのデータしか存在しない場合、足りないデータを対称位置にあるデータで置き換えて演算する」といった処理になる。

【0060】

この場合、本発明におけるLifting演算ユニットの内部構成の変更を必要とする。

【0061】

図11、12、13に前記変更後の内部構成を示す。変更後の構成は以下に対応する3種類となる。

【0062】

構成1 始端境界処理が可能なLifting演算ユニット（始端対応演算ユニット）

構成2 終端境界処理が可能なLifting演算ユニット（終端対応演算ユニット）

構成3 始端境界処理と終端境界処理の両方が可能なLifting演算ユニット（両端対応演算ユニット）

図11に示す始端境界処理が可能なLifting演算ユニットは、図14に示す左端境界部分の●で示した格子点データを演算することができる。図11で追加された2つのセレクタ1121と1123は通常処理では端子aを選択する。尚、図11～図13に示すCは、Lifting係数： α 、 β 、 γ 、 δ のいずれかである。

【0063】

境界部のデータを演算する場合、まず、セレクタ1121で端子bを選択することで、端子1103から入力するデータをそのままの値でレジスタ1113に格納する。次のサイクルで、セレクタ1123の端子bを選択することによって、端子1101から入力したデータを $2 \cdot C$ 倍した値を前記レジスタの出力に加算する。

【0064】

例えば、端子1103から入力したデータX0がレジスタ1113にそのままの値で格納され、次のサイクルで、端子1101にD1が入力されると $X0 + 2 \cdot \beta \cdot D1 (= E0)$ が端子1131から出力される。これらの制御は、セレクタ1121、1123に与える制御信号で行なう。

【0065】

図12に示す終端境界処理が可能なLifting演算ユニットは、図14に示す右端境界部分の●で示した格子点データを演算することができる。前記始端境界処理と異なり、終端境界処理ではレジスタ1213への格納時に、端子1201から入力したデータを $2 \cdot C$ 倍した値を端子1203から入力したデータに加算するよう制御する。その代わり、該レジスタ1213の出力へは何も加算しない。

【0066】

上記Lifting演算ユニットを図16のように接続した構成で、図14に示すLifting格子構造の格子点データをすべて演算することができる。Lifting格子構造は図14だけではなく、図15のような場合もある。図15の境界データを処理するには、図16の構成において、始端対応演算ユニットと終端対応演算ユニットを入れ替えればよい。しかし、図14と図15の両方を処理できるようにするには、図13に示す構成の両端対応演算ユニットを図17のように従属接続した構成が必須となる。

【0067】

順方向と逆方向の両方のウェーブレット変換が可能なLifting演算ユニットとして、図18に示す構成も考えられる。これは、図6に示した演算ユニット621に乗算器1853を付加し、セレクタ1855で、元からある乗算器1851の出力から追加した乗算器1853の出力へ切り換えることで、逆方向のウェーブレット変換に使用できるLifting演算ユニットとなる。

【0068】

つまり、Lifting演算ユニットは、乗算係数（Lifting係数）が α である乗算器1851を乗算係数（Lifting係数）が $-\delta$ の乗算器1853へ切り換え、同様にして、図6に示した演算ユニット622～623それぞれに対応する不図示のLifting演算ユニットにおいては、 β は $-\gamma$ に、 γ は $-\beta$ に、 δ は $-\alpha$ に切り換えることで、図6の構成は図8の構成と等価になるため、逆方向のウェーブレット変換ができるようになる。

【0069】

上記のような乗算係数の異なる乗算器を2個使用する構成は、図7に示す演算ユニットにも適用でき、図19に示す格子点データ演算ユニットとなる。該演算ユニットを用いた場合にも、順方向と逆方向の両方のウェーブレット変換を行なうことができる。

【0070】

以上説明したように、実施形態1によれば、Lifting格子構造上の格子点データを演算する過程で生成される中間データをレジスタに保存し、この中間

データを用いた演算を可能にすることで、入力されたデータストリームを先頭から2画素ずつペアにしてウェーブレット変換処理を行うことができる。また、簡単な回路構成で、画像境界部のデータを拡張することなく境界処理を実現することができる。

<実施形態2>

実施形態1では、低域・高域ウェーブレット変換係数の演算にそれぞれ、9タップと7タップのデータを用いる 9×7 フィルタの場合について説明した。

【0071】

実施形態2では、Lifting係数が2のべき乗だからなる、5タップと3タップのデータからなる 5×3 フィルタで、低域・高域ウェーブレット変換係数を演算する場合の構成を示す。

【0072】

Lifting係数が2のべき乗だけなので、乗算器を使う必要が無く、位取りを変更するだけで、Lifting係数の乗算ができる。

【0073】

3タップのデータから求める高域ウェーブレット変換係数H、5タップのデータから求める低域ウェーブレット変換係数Lはそれぞれ以下の式で演算する。

【0074】

$$H_{2n+1} = X_{2n+1} - 0.5 X_{2n} - 0.5 X_{2n+2} \quad (21)$$

$$L_{2n+2} = X_{2n+2} + 0.25 H_{2n+1} + 0.25 H_{2n+3} \quad (22)$$

図20に、該ウェーブレット変換処理部の構成を示す。同図において、2001はLifting係数0.5を乗算する処理に相当する位取り変換部であり、ハードウェア上の実体は配線を1ビットずらすだけである。該構成図は水平・垂直方向いずれの処理にも共通で使えるよう演算途中のデータである中間データを蓄えるバッファを遅延器2003とした。該遅延器2003がレジスタ1段である場合は水平方向、ラインメモリである場合には垂直方向のウェーブレット変換処理部とし動作する。また、2011はLifting係数0.25を乗算する処理に相当する位取り変換部である。

【0075】

逆変換は以下の式で演算する。

【0076】

$$X_{2n+2} = L_{2n+2} - 0.25 H_{2n+1} - 0.25 H_{2n+3} \quad (23)$$

$$X_{2n+1} = H_{2n+1} + 0.5 X_{2n} + 0.5 X_{2n+2} \quad (24)$$

図21に、該ウェーブレット逆変換処理部の構成を示す。1段目のLifting係数0.5から0.25へ、2段目のLifting係数が0.25から0.5へ変わっただけで、その他は前記図20とまったく同じ構成である。

【0077】

図22に、順方向と逆方向の両方のウェーブレット変換処理が可能な構成を示す。順方向変換をする時、セレクタ2201、2203は端子a側を選択し、逆方向変換をする時、該セレクタは端子b側を選択するよう制御する。

【0078】

上記Lifting係数で、順方向と逆方向のウェーブレット変換処理をする場合、図9のようにセレクタで演算ユニット間の接続状態を切り換えることも可能ではあるが、それよりも、図22のようにLifting係数を切り換える構成としたほうが、Lifting係数乗算後の値を切り換えるセレクタを追加するだけなので、増設するセレクタの数が少ない上に加減算器を使う必要も無いのでハードウェア規模が小さくなる。

【0079】

上記3種類の構成は、上述の図7にも適用できる。すなわち、図23の構成で順方向のウェーブレット変換処理ができ、図24の構成で順方向と逆方向の両方のウェーブレット変換処理ができる。

【0080】

以上説明したように、実施形態2によれば、Lifting係数が2のべき乗だけからなる場合において、Lifting格子構造上の格子点データを演算する過程で生成される中間データを遅延器を用いて、この中間データを用いた演算を可能にすることで、入力されたデータストリームを先頭から2画素ずつペアにしてウェーブレット変換処理を行うことができる。

<実施形態3>

実施形態3では、可逆（ロスレス）変換と非可逆（ロッシー）変換の切り換えが容易な構成について示す。変換処理して得られた変換係数をそのまま逆変換処理した時に元のデータを完全に復元できるような変換・逆変換を可逆変換という。画像データはデータ量が多いので、通常は、圧縮率が高くできる非可逆圧縮を用いている。しかし、診断に用いる医療用の画像など一部の画像では画像劣化の無い可逆圧縮（可逆変換）が求められる。

【0081】

図20～図24のいずれの構成においても、丸め処理器を追加することにより、通常の非可逆変換を可逆変換に変更することができる。

【0082】

丸め処理を導入することにより可逆変換が実現できる理由を簡単に述べる。

【0083】

通常は、実数計算による演算誤差が発生し、その誤差が累積することによって、量子化しない（量子化誤差が無い）場合でも、逆変換後に元と同じ値にならないことがしばしばある。

【0084】

しかし、丸め処理により整数化を行なう場合、該丸め処理が誤差の発生源となり、丸め処理が一意に定義されていて、順方向の変換処理と逆方向の変換処理で整合がとれていれば、誤差は打ち消されて無くなるので、逆変換後に元の値に戻ることが理論的に保証されるわけである。

【0085】

図22と図23に対して、丸め処理器を追加した構成を、図25、図26に示す。入力データを整数データと仮定すると、この場合の丸め処理は、演算処理によって発生する小数部を整数化することである。

【0086】

丸め処理の方法としては、四捨五入、切り捨て、切り上げなどいろいろあるが、図25における初段の丸め処理器2501、2段目の丸め処理器2503いずれも、順方向変換時は四捨五入を行なう。逆方向変換時には、初段の丸め処理器2501は0.75以上を切り上げて、0.5以下を切り捨て処理し、2段目の

丸め処理器2503は切り捨てを行なう。

【0087】

図26における初段の丸め処理器2601は順方向変換時に切り捨てを行ない、逆方向変換時に四捨五入を行なう。2段目の丸め処理器2603は順方向変換時に四捨五入を行ない、逆方向変換時に切り捨て処理を行なう。

【0088】

よって、各丸め処理器は変換処理の種類に対応して、丸め処理の内容を切り換えなければならないため、その切り換えのための制御信号を入力している。

【0089】

上記丸め処理器の前段の加算器や減算器でオフセットを加算することで、該丸め処理器での処理を切り捨てだけにすることもできる。ここで言う切り捨てとは小数部の切り捨てのことと言っている。

【0090】

図25、図26に対してオフセットを加算する場合の構成を図27と図28に示す。同図における丸め処理器は上述のごとく小数部の切り捨てだけをする。加算するオフセット値は変換の種類（順方向・逆方向）によって異なり、図27におけるオフセット生成器2701は順方向変換時には0.5を、逆方向変換時には0.25を出力し、もう1つのオフセット生成器2703は順方向変換時には0.5を、逆方向変換時には0を出力する。

【0091】

図28におけるオフセット生成器2801は順方向変換時には0を、逆方向変換時には2を出力し、もう1つのオフセット生成器2803は順方向変換時には2を、逆方向変換時には0を出力する。

【0092】

いずれのオフセット生成器も変換処理の種類に対応して、出力するオフセット値を切り換えなければならないため、その切り換えのための制御信号を入力している。逆に、丸め処理は一律に小数部を切り捨てるだけであるから、制御信号を入力する必要がない。この場合の丸め処理器の物理的な実体は無く、小数部の信号の配線が断線状態になっている。

【0093】

小数部の切り捨てで丸め処理を行なう図27、図28の構成から、次のような新たな構成も考えられる。

【0094】

丸め処理器を、該小数部の信号を制御信号でマスクする構成に変えることにより、該制御信号で丸め処理のON/OFF制御ができ、可逆変換と非可逆変換の切り換えが容易に行なえるようになる。もちろん、丸め処理のON/OFFだけでなく、加算器・減算器に入力するオフセットも連動してON/OFFする必要がある。

【0095】

図29、図30に示す構成は、図27、図28の構成に対して、可逆／非可逆変換を切り換える制御信号をあらたに設け、該制御信号によって、可逆変換時には丸め処理器において小数部をマスクし、非可逆変換時にはオフセット生成器においてオフセット出力をマスクする。

【0096】

上述した制御により、オフセットの加算と小数部の切り捨てを行なう可逆変換処理と、オフセットの加算も小数部の切り捨ても無い非可逆変換処理の切り換えが容易に行なえるウェーブレット変換装置が構成できるようになった。

【0097】

また、図25、図26において、可逆／非可逆変換を切り換える制御信号により丸め処理器をパスするようにセレクタを設けることで、該可逆／非可逆変換を切り換えることも可能である。

【0098】

以上説明したように、実施形態3によれば、実施形態2で説明した効果に加えて、実施形態2で説明した構成に対し丸め処理器、オフセット生成器を追加した簡単な回路構成で、ウェーブレット変換における可逆変換と非可逆変換の切り換えを容易に行なうことができる。

<実施形態4>

これまでの実施形態は、水平方向のウェーブレット変換もしくは垂直方向のウ

エーブレット変換のどちらか一方の処理を行なうものであった。実施形態4では水平・垂直の2次元ウェーブレット変換処理を行なう構成について示す。

【0099】

前記Lifting演算ユニット、あるいは、格子点演算ユニットを用いた2次元ウェーブレット変換処理器は図31のような構成となる。該ウェーブレット変換処理器は前述の2のべき乗のLifting係数を用いた 5×3 フィルタの処理をするものである。

【0100】

同図において、3101は、順方向変換時のデータを入力する端子である。3103は、逆方向変換時のデータを入力する端子である。3111と3113は、垂直方向のウェーブレット変換処理を行なう演算ユニットである。3121と3123は、水平方向のウェーブレット変換処理を行なう演算ユニットである。3115と3125は、変換モード（順方向変換・逆方向変換）に応じて上記演算ユニットへの入力を切り換えるセレクタである。3117と3127は、水平または垂直ウェーブレット変換されたデータを 2×2 単位で回転して、次の段のウェーブレット変換処理部に供給するデータ回転部である。3130は、順方向または逆方向2次元ウェーブレット変換されたデータを一時的に蓄えて外部のメモリ等へ出力するためのバッファである。

【0101】

順方向変換時には、端子3101から入力されたデータがセレクタ3115を経由して演算ユニット3111に供給される。該演算ユニット3111と次の演算ユニット3113とで垂直方向のウェーブレット変換処理を行ない、変換結果をデータ回転部3117へ送る。

【0102】

ここで、2つのデータ回転部3117、3127の簡単な説明を行なう。データ回転部3117、3127は図32に示すような構成で、2つ並列に入力されるデータを2サイクル周期で 2×2 単位の回転を行ない、次段のセレクタへ送る。

【0103】

図32において、データ入力端子3201と3203から並列に入力されたデータは、それぞれレジスタ3211～3215にて、1サイクル毎にシフトしながら格納される。

【0104】

m サイクル目と $m+1$ サイクル目に端子3201に入力されたデータは、 $m+2$ サイクル目にセレクタ3221と3223を経由して端子3231と3233から出力される。

【0105】

端子3203へ上記データと同時に並列入力されたデータは、 $m+3$ サイクル目に端子3231と3233から出力される。

【0106】

$m+2$ サイクル目と $m+3$ サイクル目に入力された 2×2 のデータは $m+4$ 、 $m+5$ サイクル目に出力される。

【0107】

データ回転部3117では、垂直方向2サンプルの並列データが水平方向2サンプルの並列データに変換され、データ回転部3127では、水平方向2サンプルの並列データが垂直方向2サンプルの並列データに変換される。

【0108】

データ回転部3117によって変換された、水平方向2サンプルの並列データはセレクタ3125を経由して、演算ユニット3121に入力される。演算ユニット3121、3123からなる水平ウェーブレット変換部は該入力データを変換処理し、変換結果を演算ユニット3123からバッファ3130へ出力する。上記水平ウェーブレット変換部は、垂直方向の低域ウェーブレット変換係数と高域ウェーブレット変換係数を交互に処理するため、前記演算ユニット3121、3123内の各遅延器は2段のレジスタからなる。

【0109】

逆方向変換時には、端子3103から入力されたデータがセレクタ3125を経由して演算ユニット3121に供給される。不図示の制御信号により、逆方向ウェーブレット変換処理モードとなった演算ユニット3121と次の演算ユニッ

ト3123とで水平方向の逆ウェーブレット変換処理を行ない、変換結果をデータ回転部3127へ送る。

【0110】

逆方向変換時の水平ウェーブレット変換部におけるデータの処理順序は、順方向変換時に出力するデータをそのまま逆に処理すればよいが、この順序は通常の1ライン単位の処理とは異なるので、少し補足説明する。

【0111】

実施形態4の構成では2ライン単位で処理をする。但し、2ラインを同時に並列処理することはできないので、ライン交互に水平2サンプルを入力して、逆方向の水平ウェーブレット変換処理を行なう。演算ユニット内の各遅延器はレジスタ2段で構成されているので、2ラインを交互に処理することが可能である。逆変換処理した結果も当然2ライン交互に出力される。それを 2×2 のデータ回転処理部3127で2ライン並列のデータに変換することで、垂直方向のウェーブレット変換処理部に都合のよいデータを送ることができる。

【0112】

その後、セレクタ3115で上記データを選択して演算ユニット3111に入力すれば、垂直方向の逆ウェーブレット変換処理が行なわれ、変換結果が演算ユニット3113からバッファ3130へ出力される。

【0113】

9×7 フィルタで処理する場合には、垂直処理部の演算ユニット3111、3113並びに水平処理部の演算ユニット3121、3123を、それぞれ図9に示す構成に置き換えればよい。

【0114】

以上説明したように、実施形態4によれば、実施形態1～実施形態3で説明した構成を利用して、簡単な回路構成で、入力されたデータストリームを先頭から2ラインずつ2次元ウェーブレット変換処理を行うことができる。

【0115】

尚、上述した本発明で実現されるハードウェア構成は、専用ハードウェア基板あるいは専用ハードウェアチップとしてパーソナルコンピュータ等の端末に実装

され、その端末が有するC P U等の制御部が上述した制御あるいは制御信号の生成を行うことにより、該ハードウェア構成による処理が実行される。

【0116】

また、本発明で実現されるハードウェア構成による処理をソフトウェアとして端末の記憶装置に記憶しておき、このソフトウェアを端末に搭載されるC P Uによって実行させることも可能である。このソフトウェアが実行する処理について、以下、図32に示す。

【0117】

図32は本発明で実行される処理を示すフローチャートである。

【0118】

まず、ステップS101で、データを入力する。次に、ステップS102で、上述の(9)～(12)式を用いて、Liftинг格子構造上の格子点データを演算する途中のデータである中間データを生成する。次に、ステップS103で、生成した中間データを保持する。次に、ステップS104で、上述の(13)～(16)式を用いて、保持した中間データから格子点データを生成する。

【0119】

ステップS105で、処理対象のデータが存在するか否かを判定する。処理対象のデータが存在する場合（ステップS105でYES）、ステップS106に進み、次の処理対象のデータを入力し、ステップS102に戻る。一方、処理対象のデータが存在しない場合（ステップS105でNO）、処理を終了する。

【0120】

尚、本発明は、複数の機器（例えばホストコンピュータ、インターフェース機器、リーダ、プリンタなど）から構成されるシステムに適用しても、一つの機器からなる装置（例えば、複写機、ファクシミリ装置など）に適用してもよい。

【0121】

また、本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ（またはC P UやM P U）が記憶媒体に格納されたプログラムコードを読み出し実行することによっても、達成されることは言

うまでもない。

【0122】

この場合、記憶媒体から読み出されたプログラムコード 자체が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。

【0123】

プログラムコードを供給するための記憶媒体としては、例えば、フロッピディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD-R／RW、DVD-ROM／RAM、磁気テープ、不揮発性のメモリカード、ROMなどを用いることができる。

【0124】

また、コンピュータが読み出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているOS（オペレーティングシステム）などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0125】

更に、記憶媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0126】

本発明を上記記憶媒体に適用する場合、その記憶媒体には、先に説明した図3に示すフローチャートに対応するプログラムコードが格納されることになる。

【0127】

【発明の効果】

以上説明したように、本発明によれば、効率的にウェーブレット変換処理を実

行できるデータ処理装置及びその制御方法、データ処理方法、コンピュータ可読メモリを提供できる。

【図面の簡単な説明】

【図1】

順変換のLifting Schemeを示す図である。

【図2】

逆変換のLifting Schemeを示す図である。

【図3】

DWT（順方向ウェーブレット変換）のLifting格子構造を示す図である。

【図4】

IDWT（逆方向ウェーブレット変換）のLifting格子構造を示す図である。

【図5】

本発明の内容をLifting格子構造で表現した図である。

【図6】

実施形態1の構成を示す図である。

【図7】

格子点データ演算ユニットの構成を示す図である。

【図8】

実施形態1の応用例であるIDWT変換装置の構成を示す図である。

【図9】

実施形態1の応用例であるDWT／IDWT変換装置の構成を示す図である。

【図10】

実施形態1の応用例であるDWT／IDWT変換装置の構成を示す図である。

【図11】

始端境界処理が可能なLifting演算ユニットの構成を示す図である。

【図12】

終端境界処理が可能なLifting演算ユニットの構成を示す図である。

【図13】

始端と終端の両方の境界処理が可能なLifting演算ユニットの構成を示す図である。

【図14】

Lifting格子構造中の境界部のデータを示す図である。

【図15】

Lifting格子構造中の境界部のデータを示す図である。

【図16】

図14の境界部のデータの処理が可能なDWT変換器を示す図である。

【図17】

図14と図15の両方の境界部のデータの処理が可能なDWT変換器を示す図である。

【図18】

DWT／IDWTが可能なLifting演算ユニットの他の構成を示す図である。

【図19】

DWT／IDWTが可能な格子点演算ユニットの他の構成を示す図である。

【図20】

5×3フィルタのDWT変換処理部の構成を示す図である。

【図21】

5×3フィルタのIDWT変換処理部の構成を示す図である。

【図22】

5×3フィルタのDWT／IDWT変換処理部の構成を示す図である。

【図23】

5×3フィルタのDWT変換処理部の他の構成を示す図である。

【図24】

5×3フィルタのDWT／IDWT変換処理部の他の構成を示す図である。

【図25】

5×3フィルタの可逆DWT／IDWT変換処理部の構成を示す図である。

【図26】

5×3 フィルタの可逆DWT／IDWT変換処理部の他の構成を示す図である

【図27】

5×3 フィルタの可逆DWT／IDWT変換処理部の他の構成を示す図である

【図28】

5×3 フィルタの可逆DWT／IDWT変換処理部の他の構成を示す図である

【図29】

5×3 フィルタの可逆／非可逆・DWT／IDWT変換処理部の構成を示す図である。

【図30】

5×3 フィルタの可逆／非可逆・DWT／IDWT変換処理部の他の構成を示す図である。

【図31】

5×3 フィルタの2次元DWT／IDWT変換装置の構成を示す図である。

【図32】

データ回転部の構成を示す図である。

【図33】

本発明で実行される処理を示すフローチャートである。

【符号の説明】

601、603、605、607 端子

611 乗算器

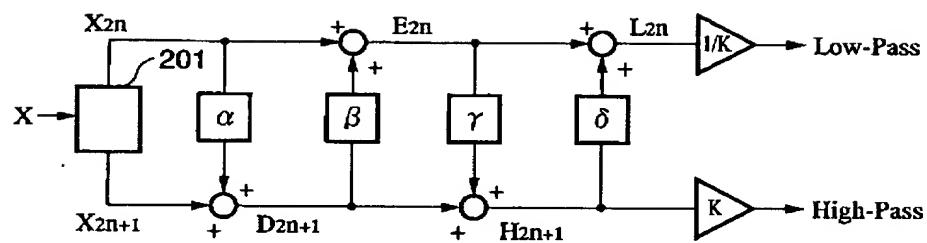
613 レジスタ

615、617 加算器

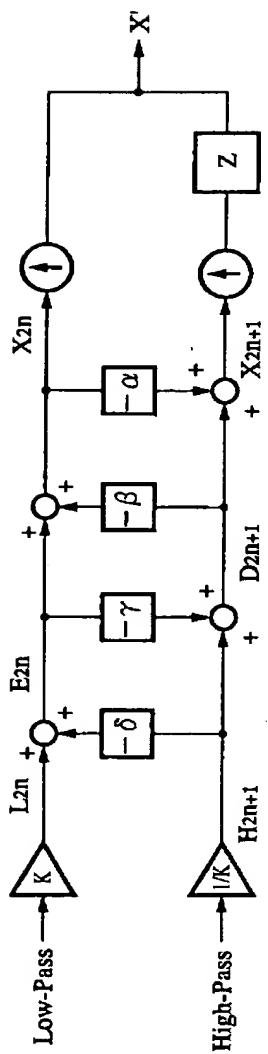
621～624 Lifting演算ユニット

【書類名】 図面

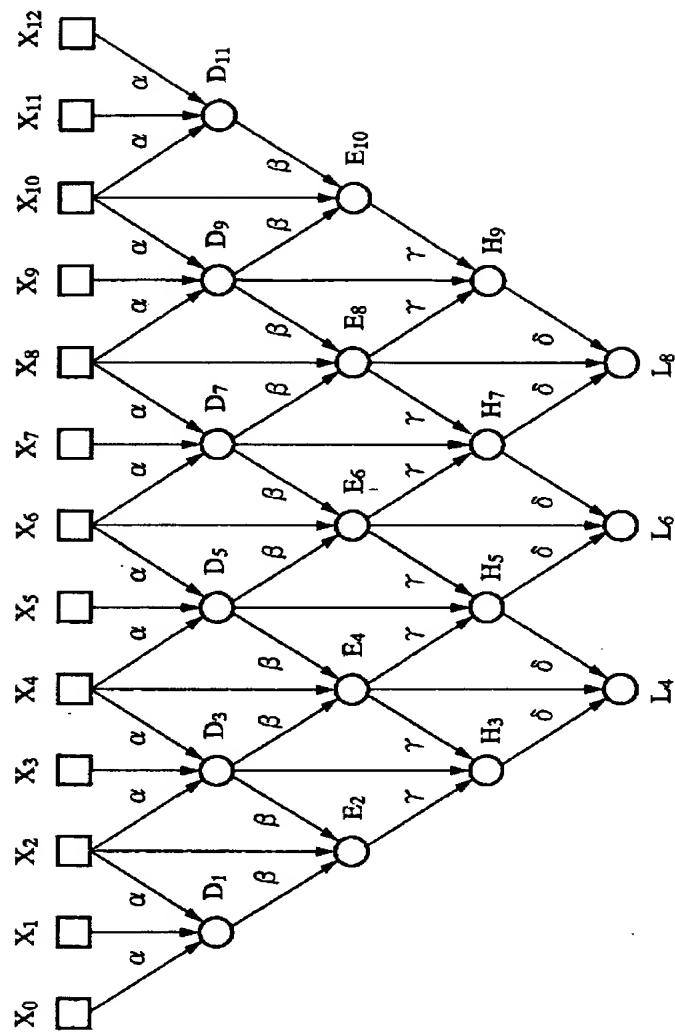
【図1】



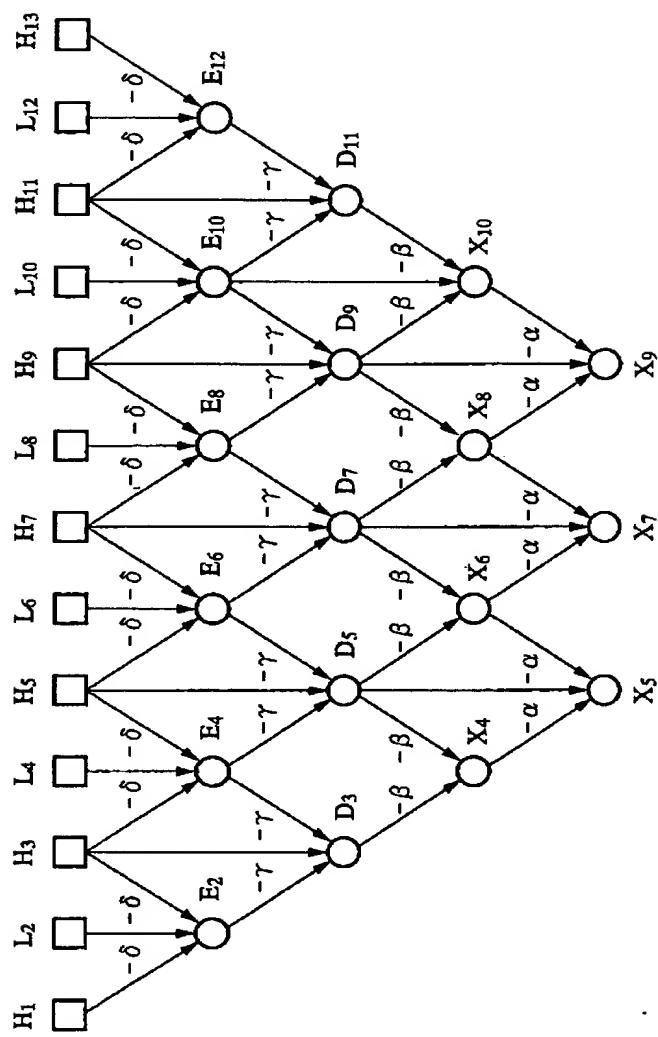
【図2】



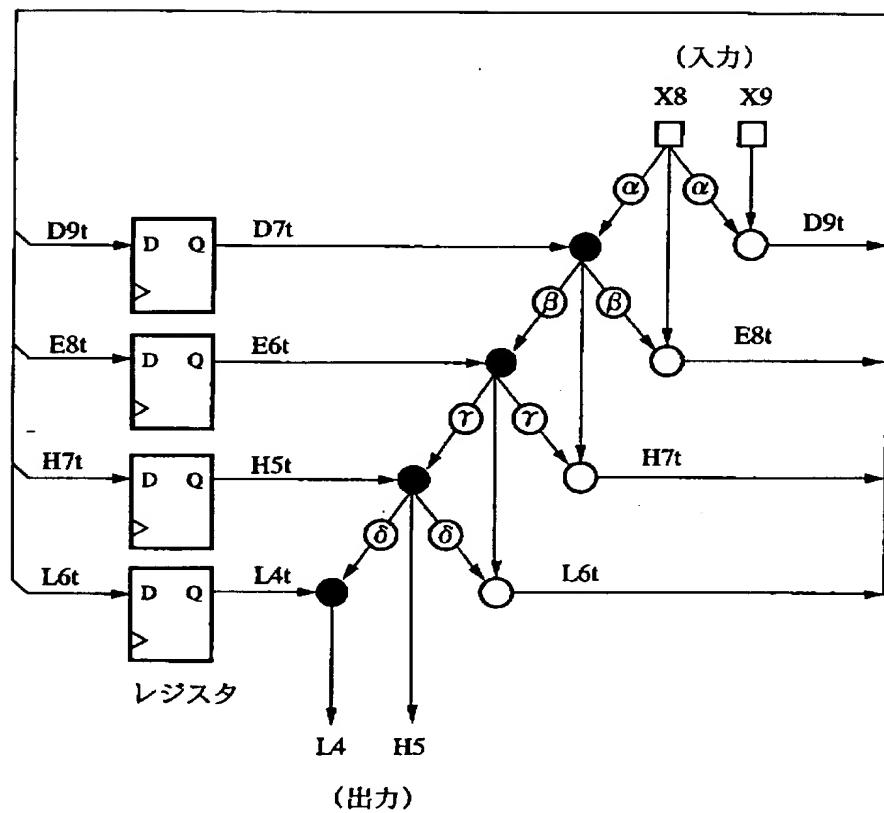
【図3】



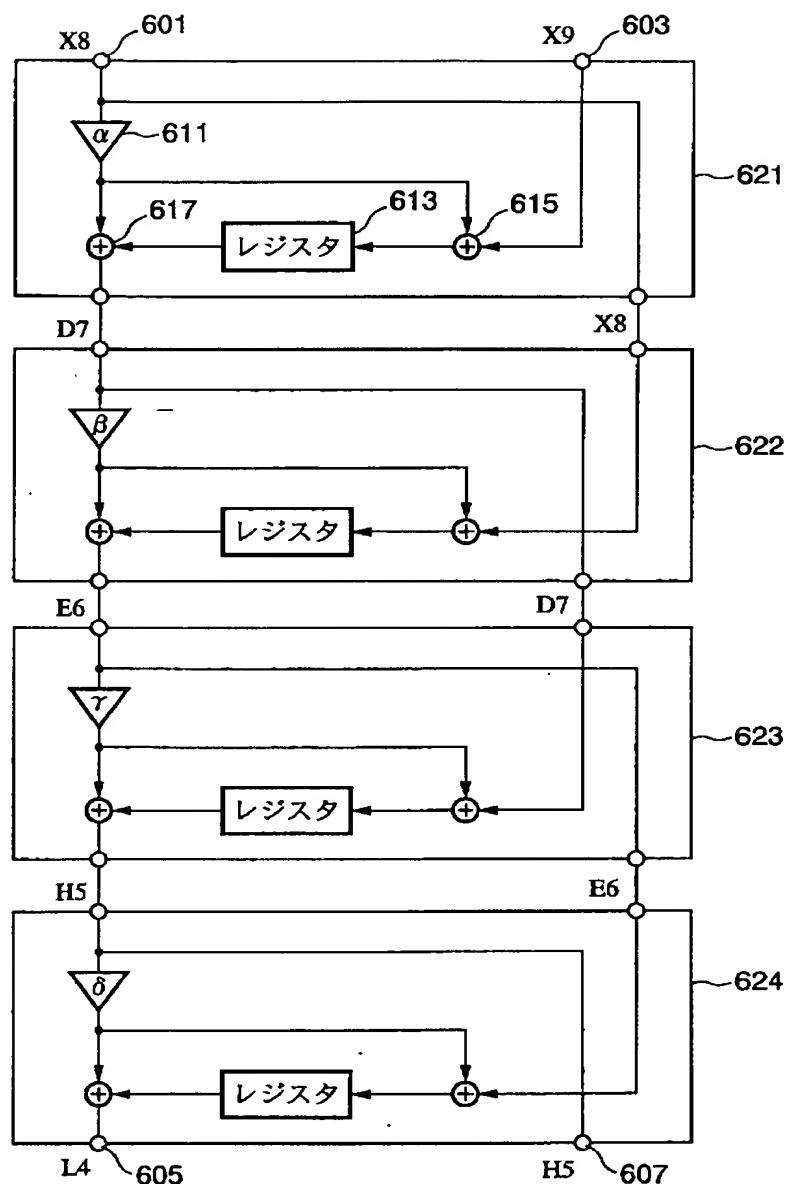
【図4】



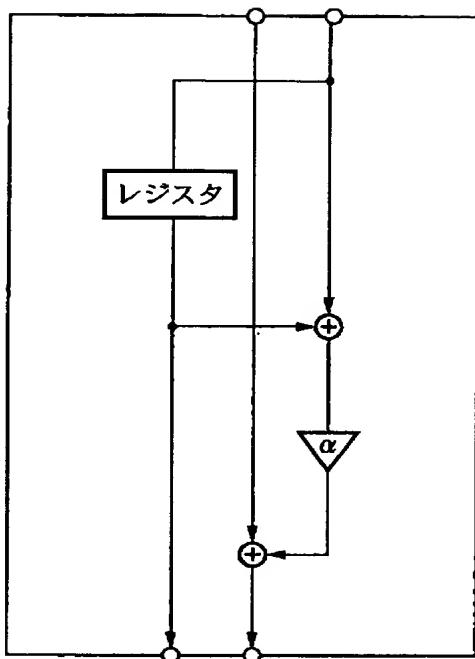
【図5】



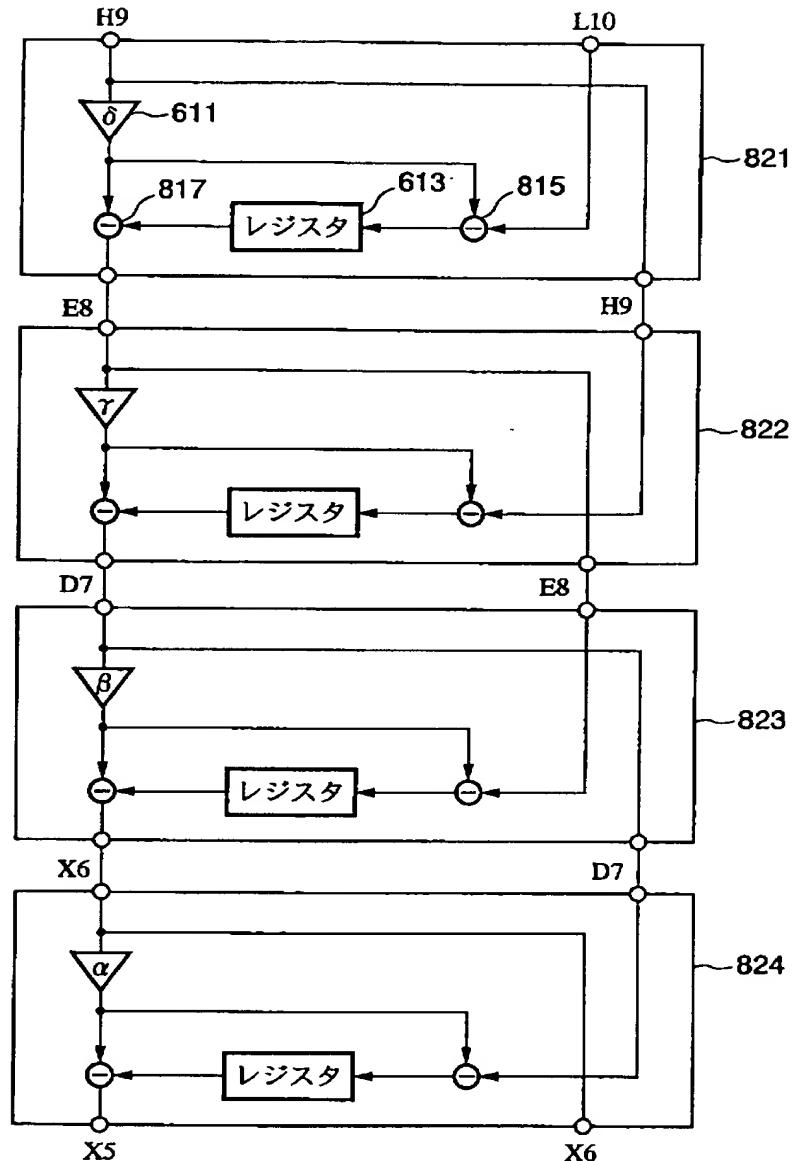
【図6】



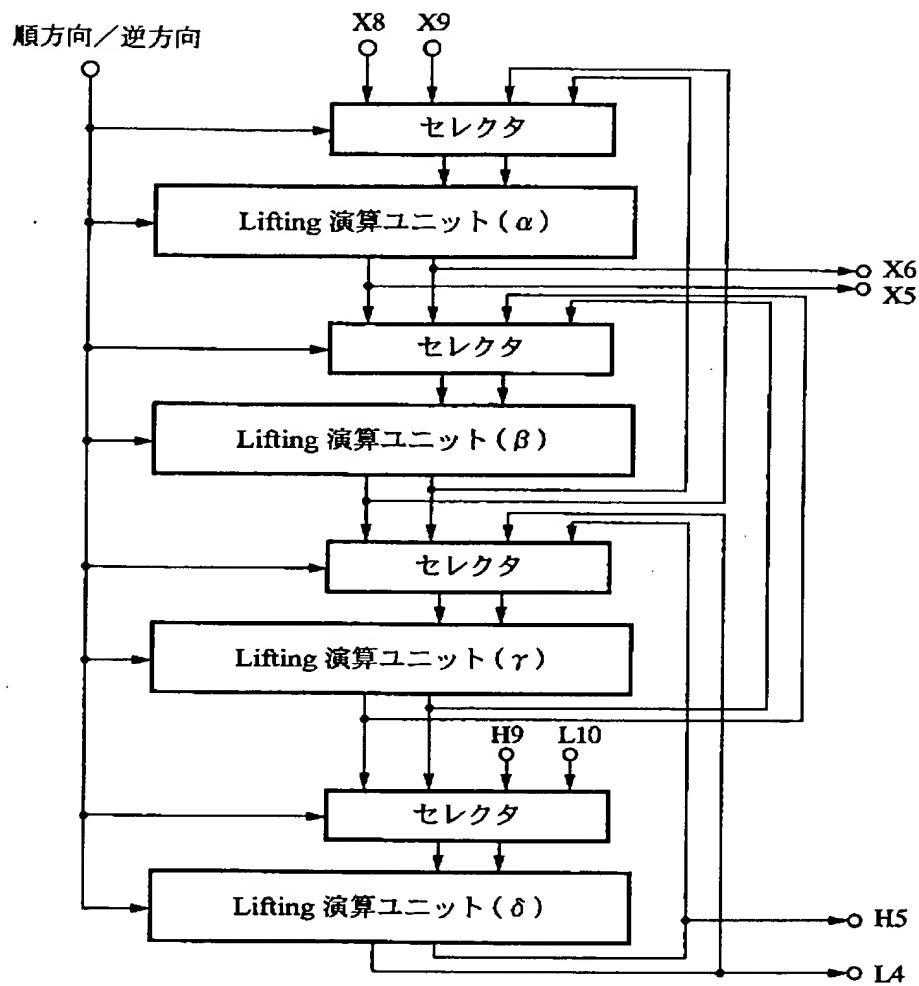
【図7】



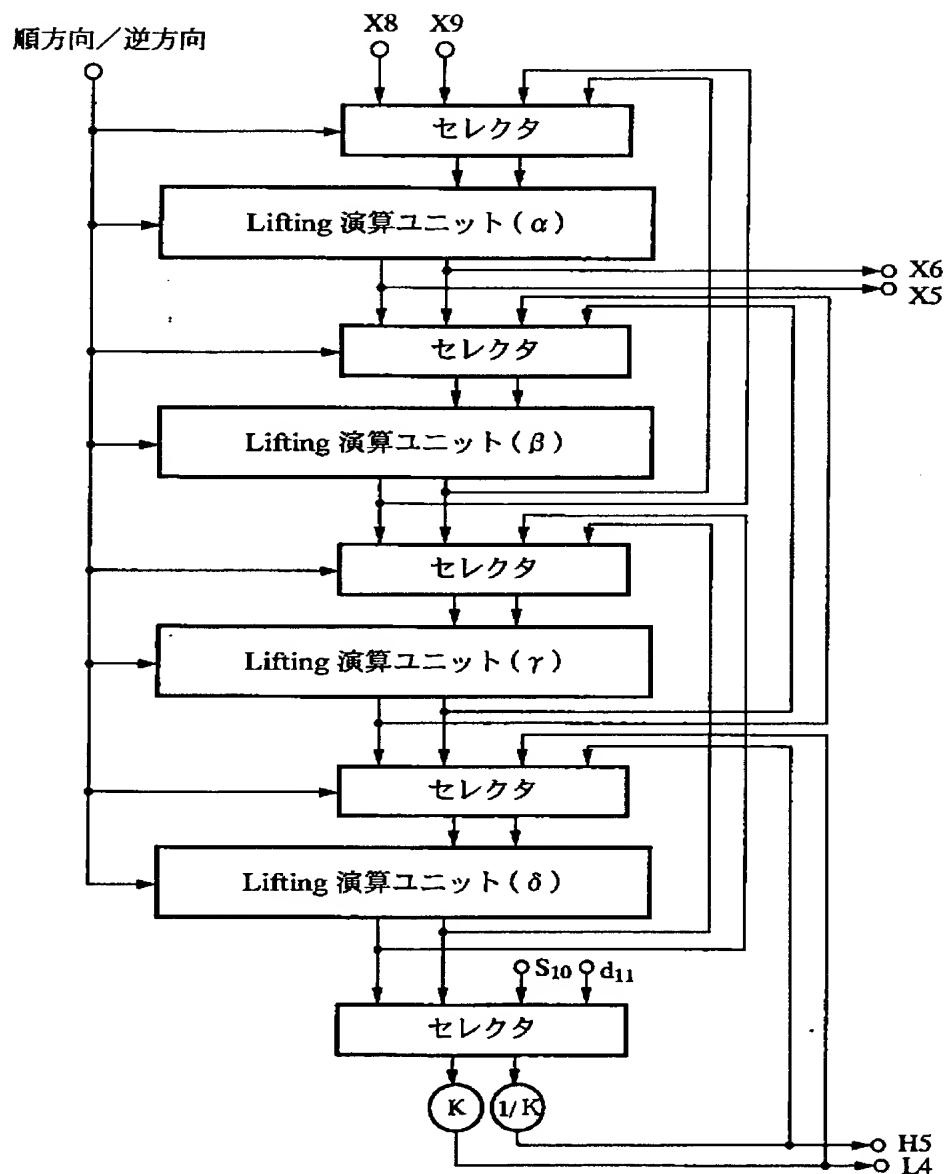
【図8】



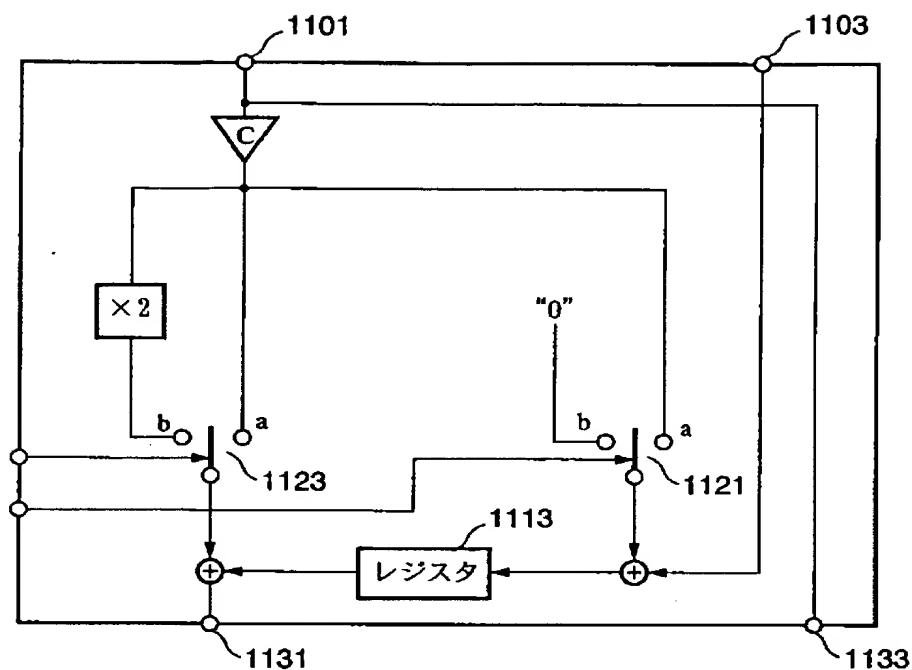
【図9】



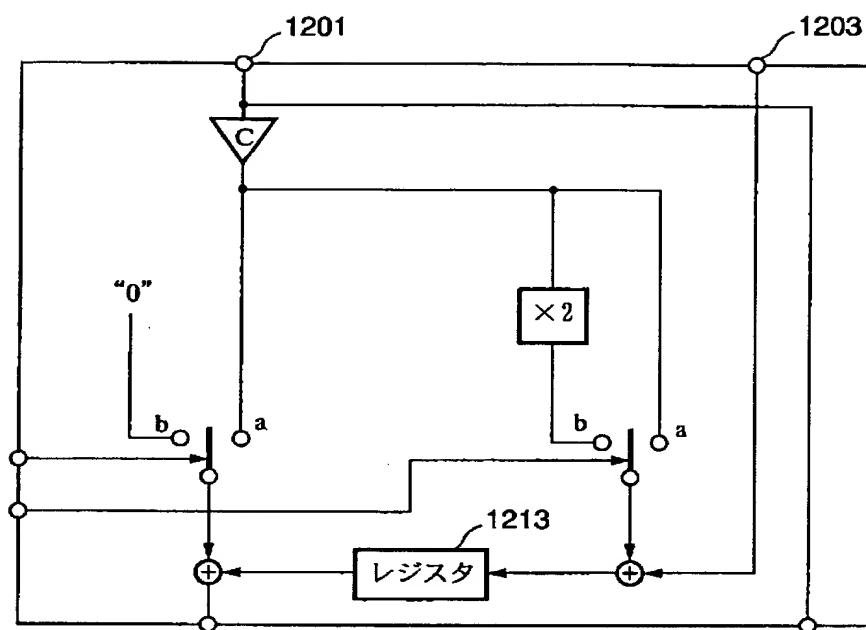
【図10】



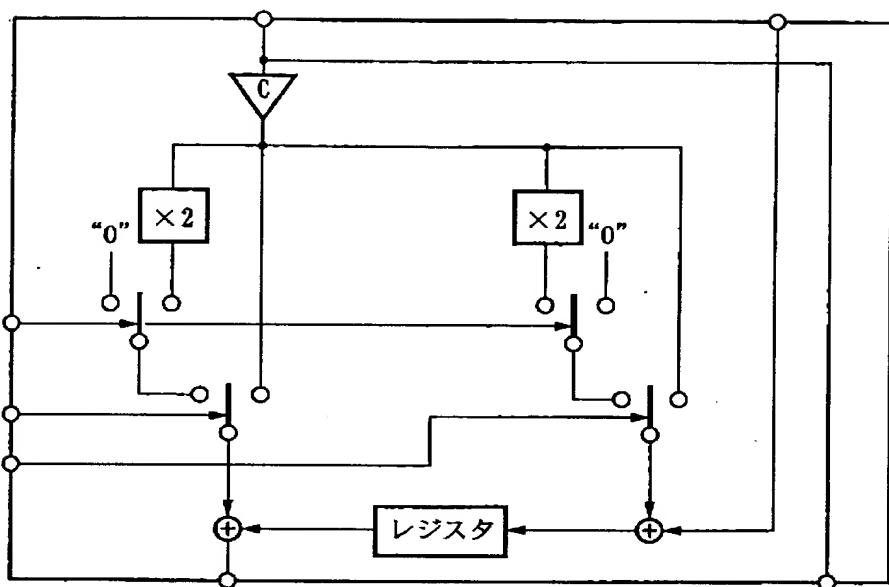
【図11】



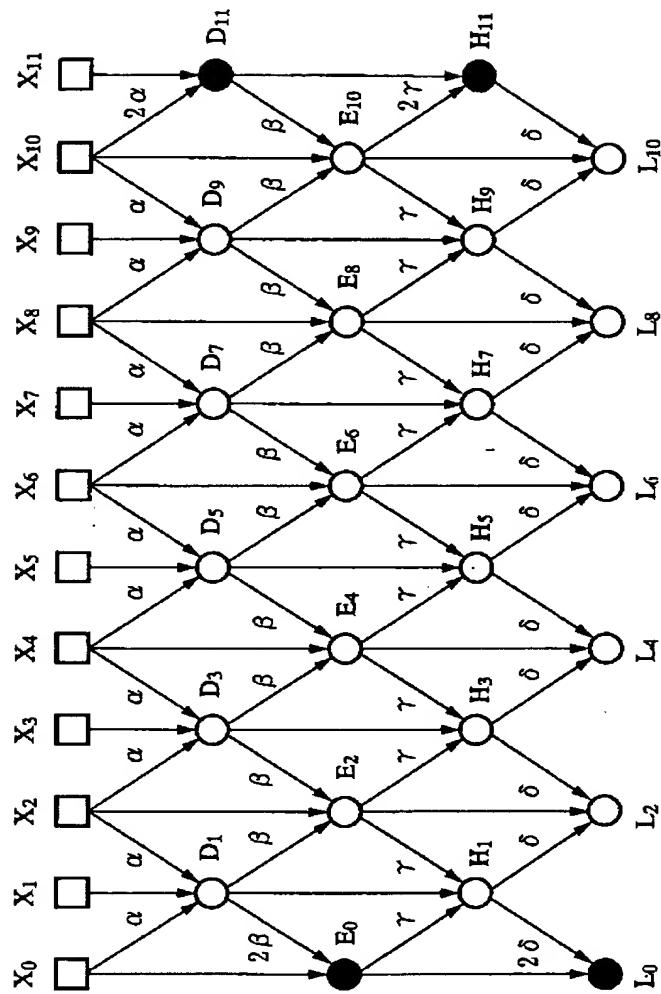
【図12】



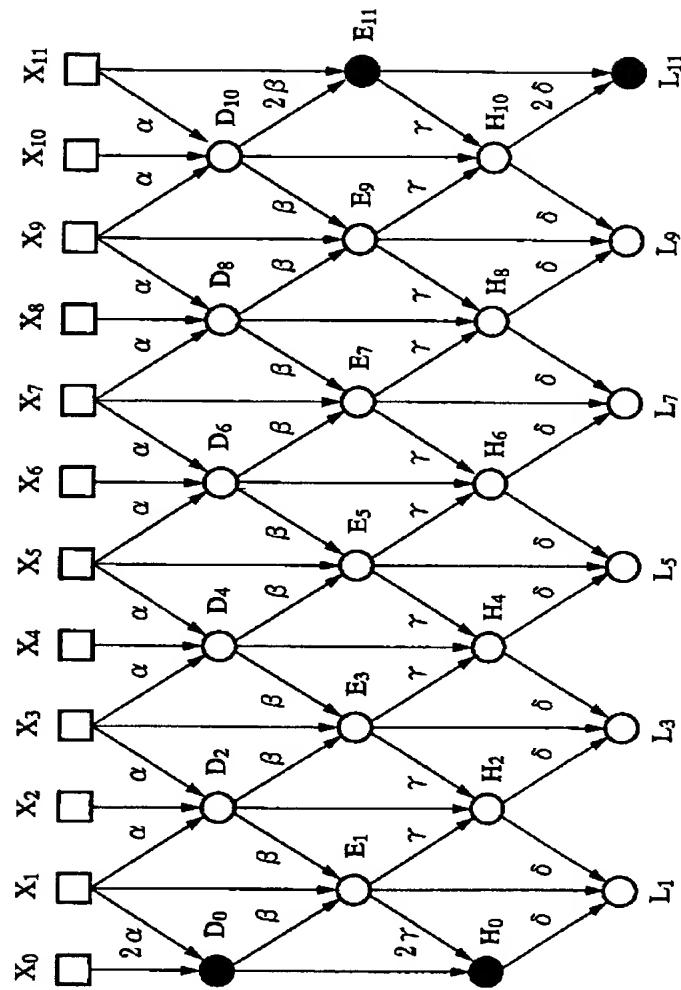
【図13】



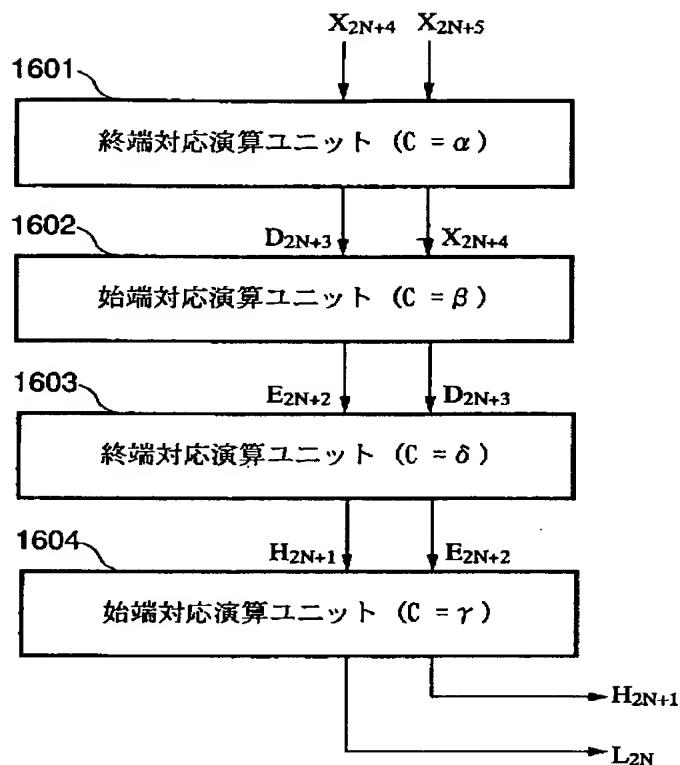
【図14】



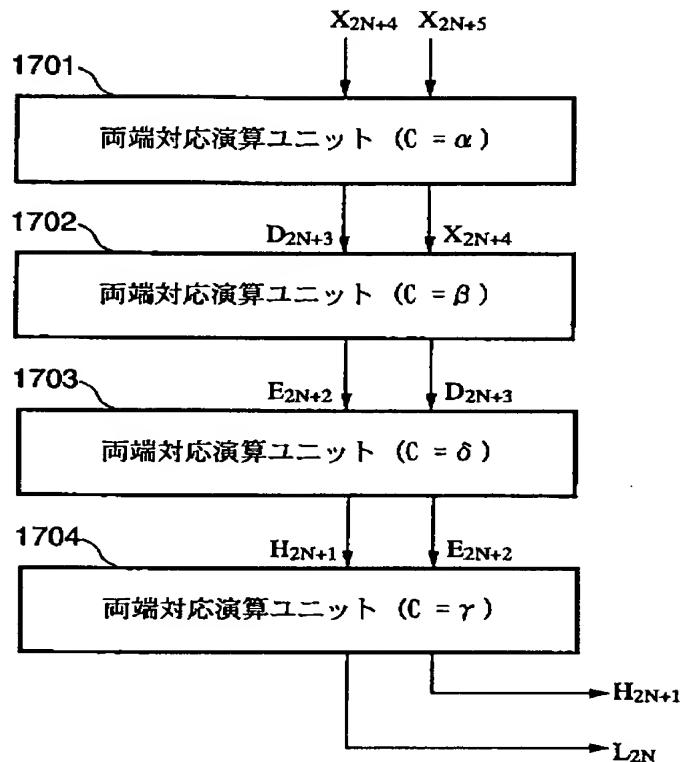
【図15】



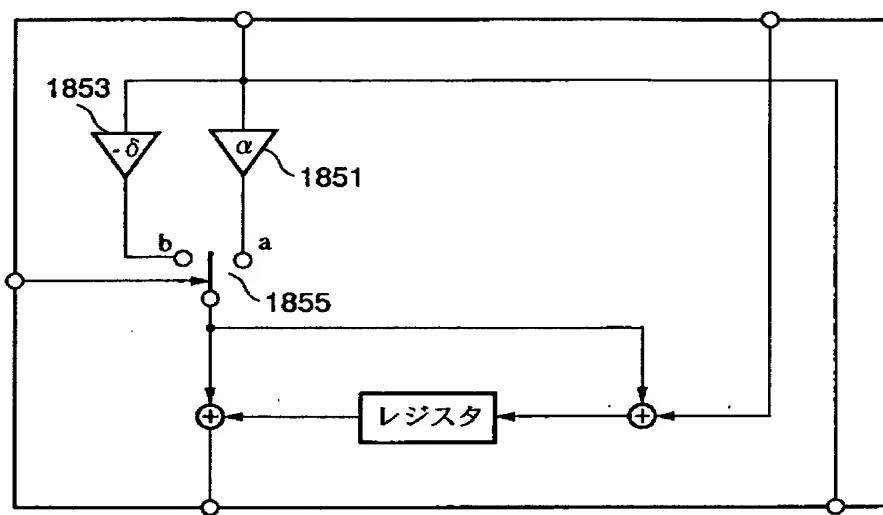
【図16】



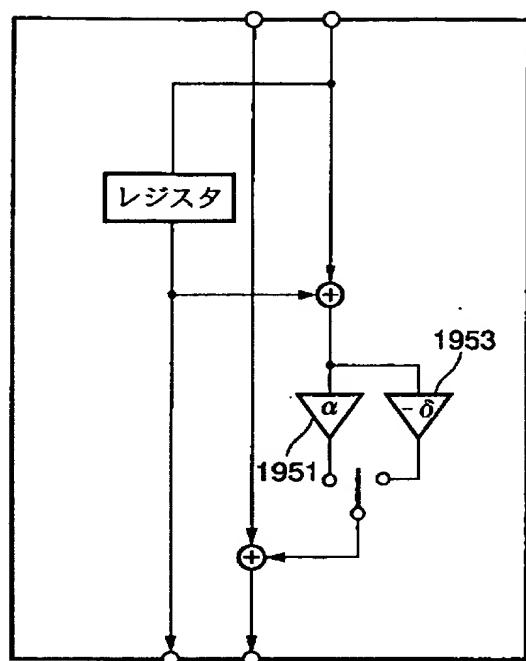
【図17】



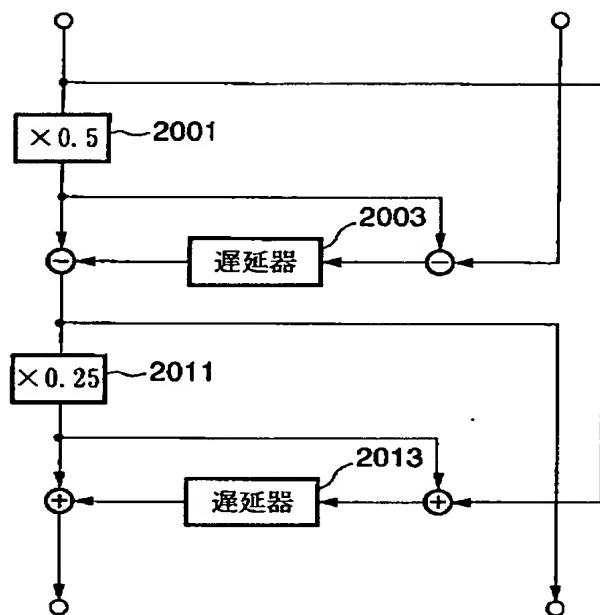
【図18】



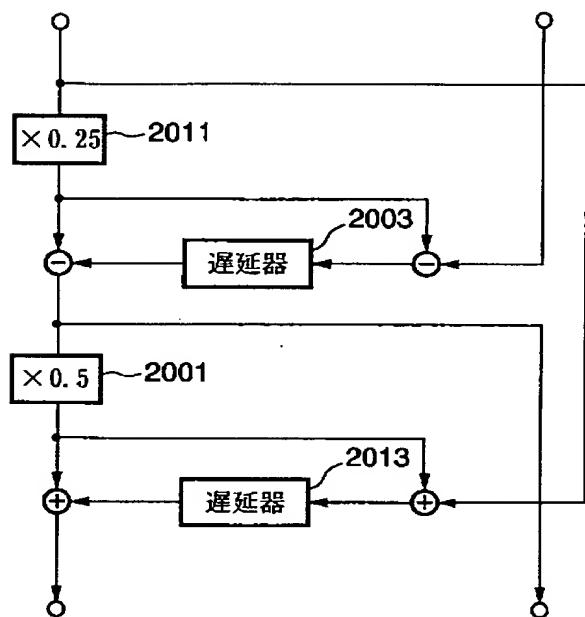
【図19】



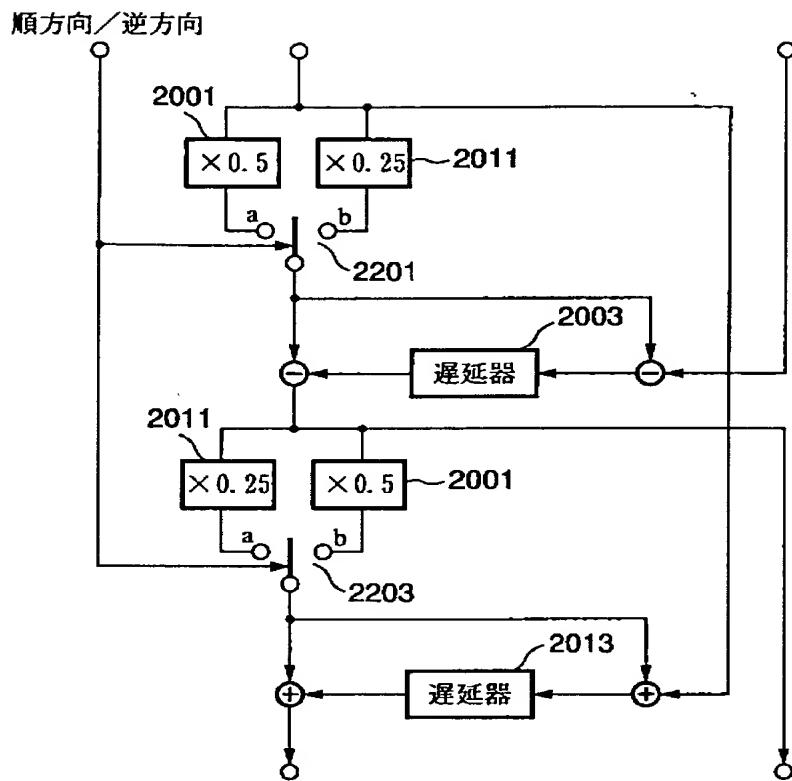
【図20】



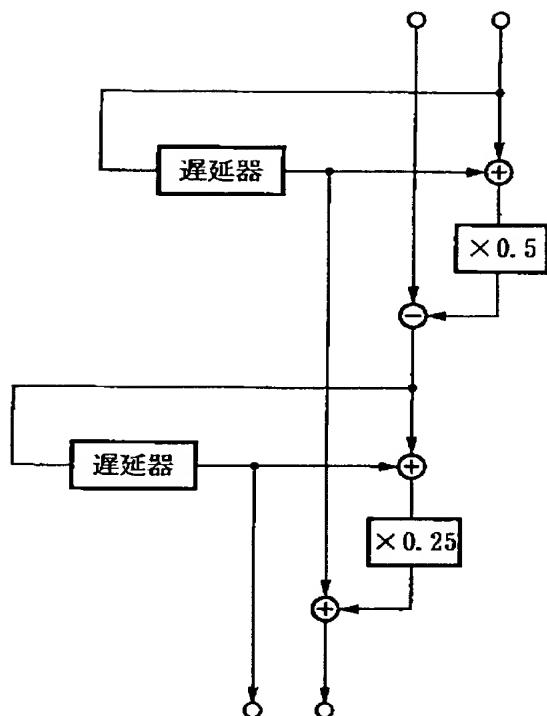
【図21】



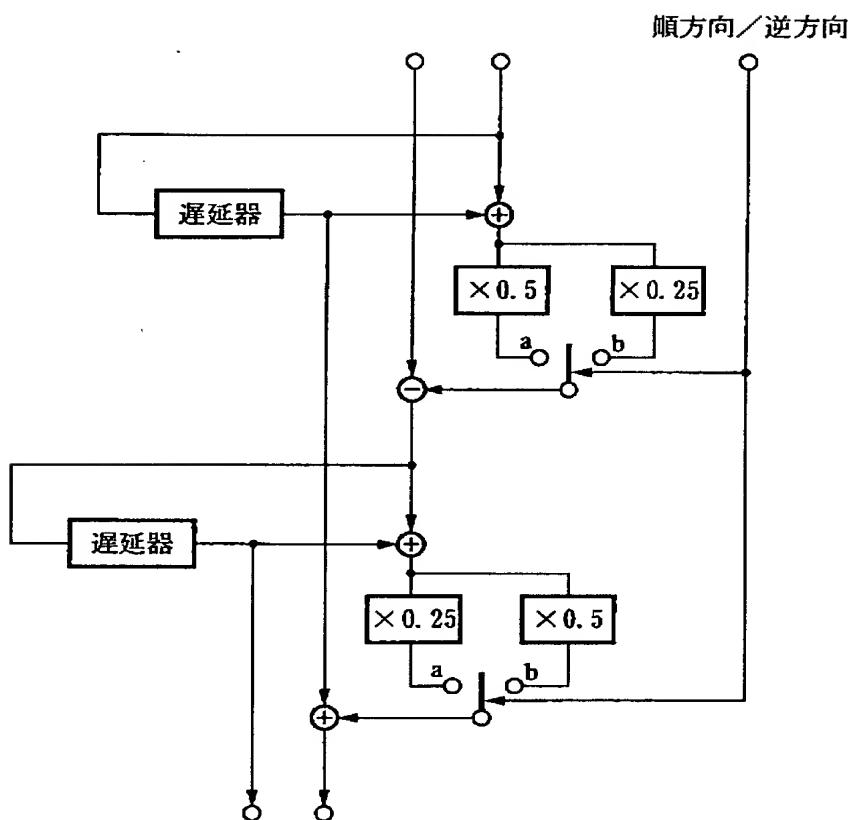
【図22】



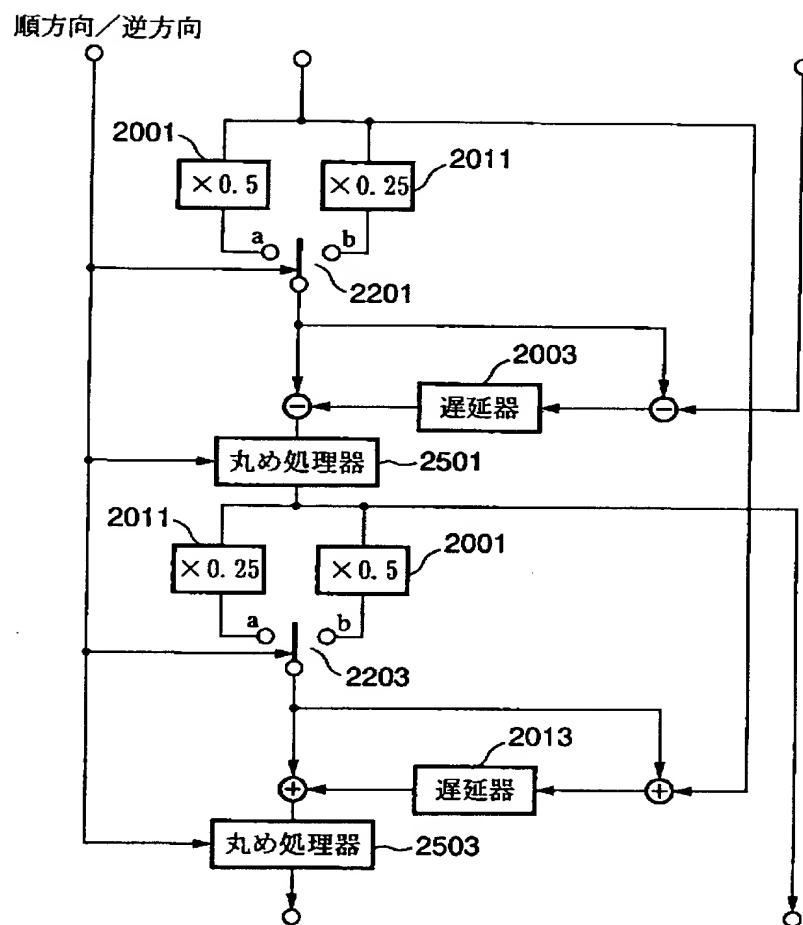
【図23】



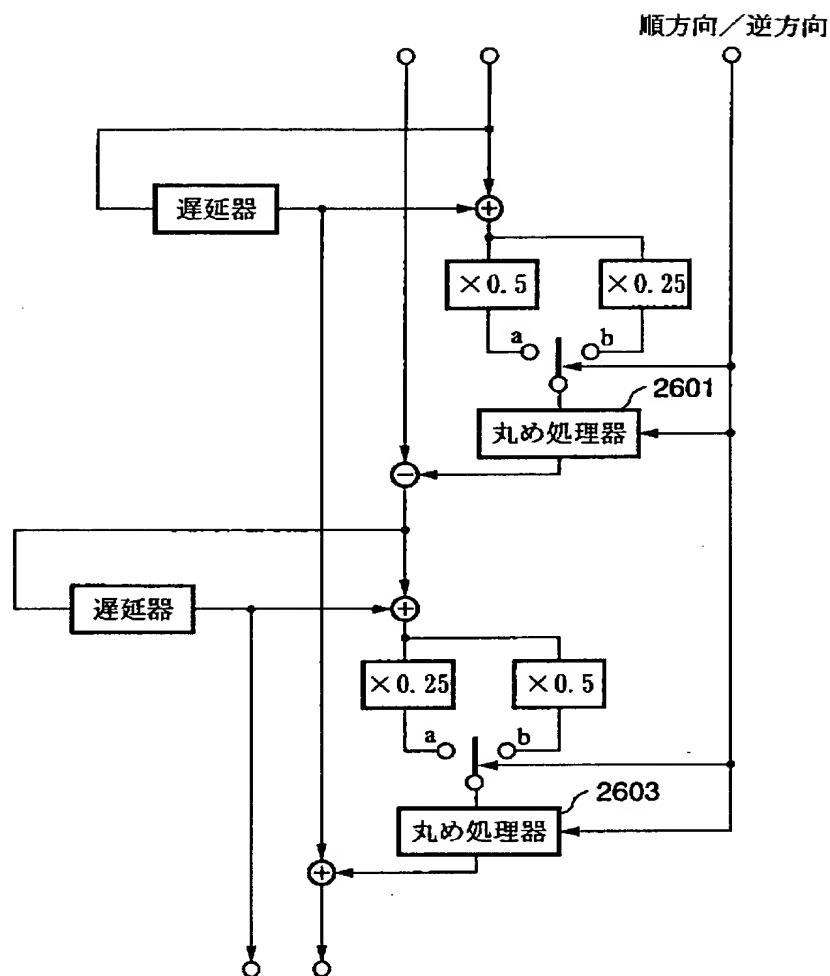
【図24】



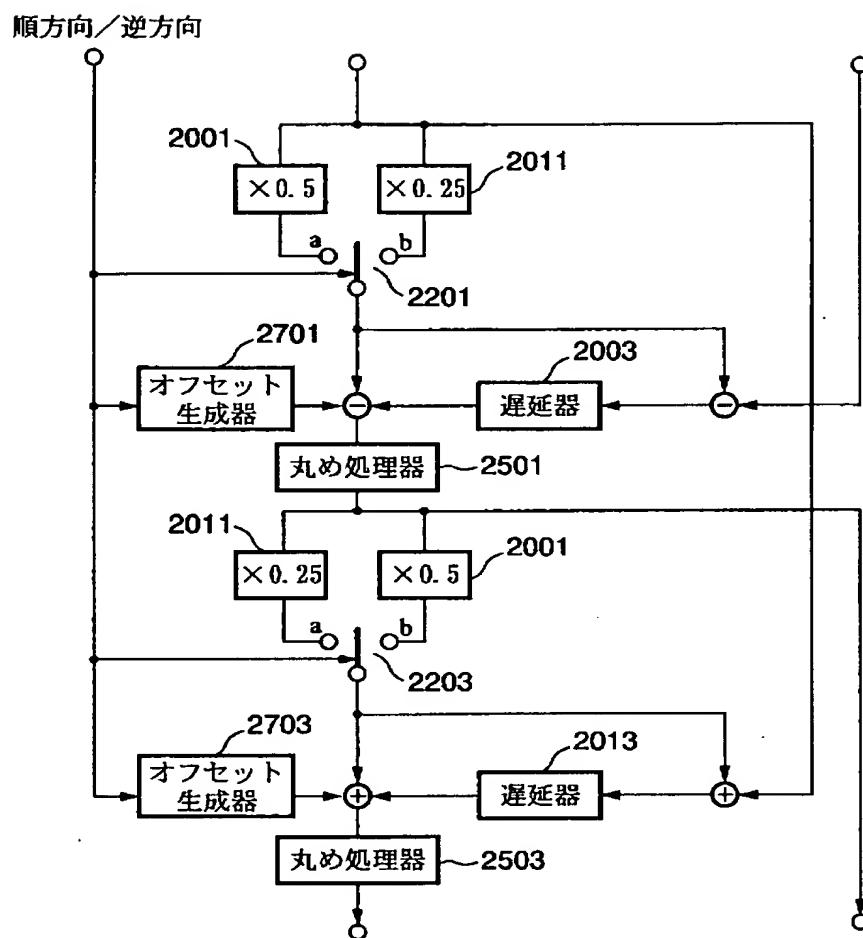
【図25】



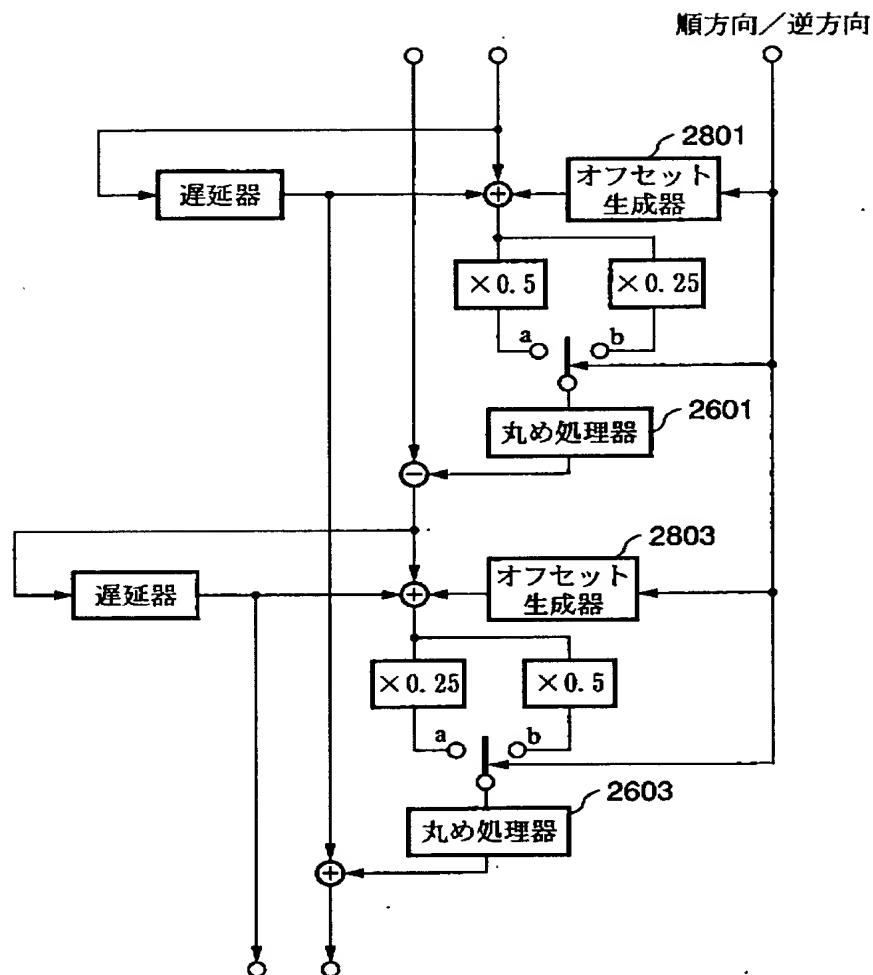
【図26】



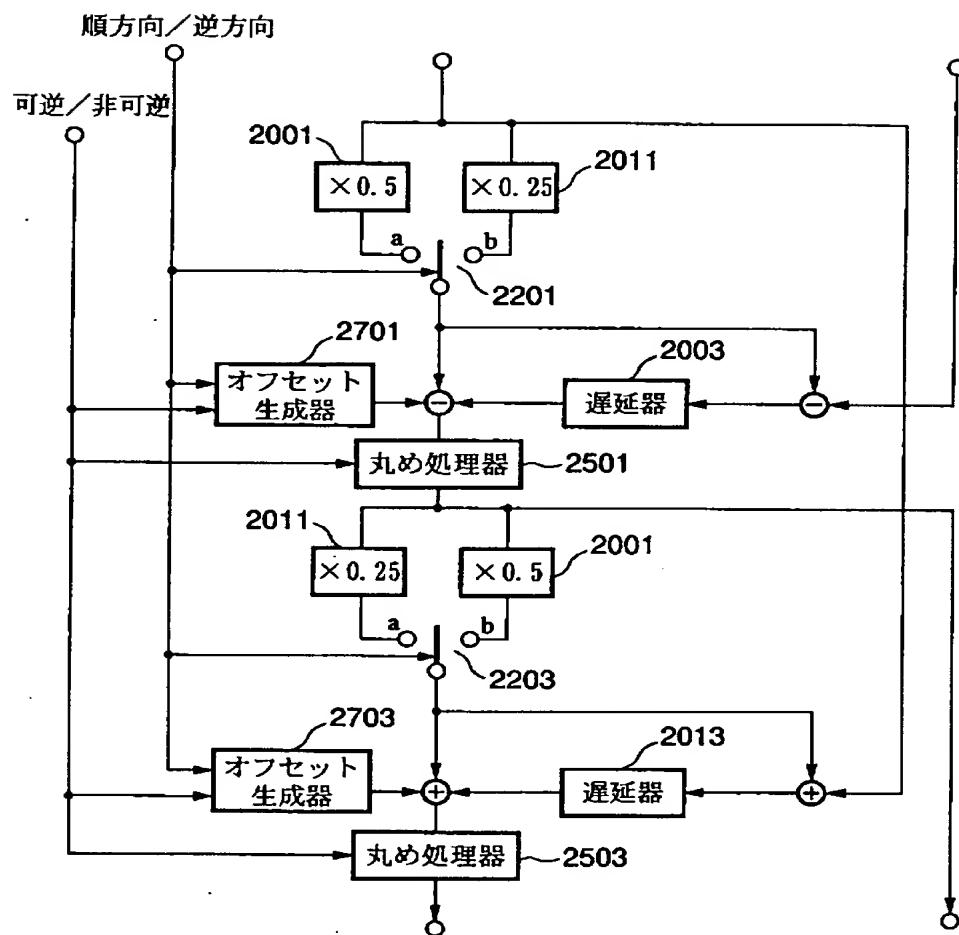
【図27】



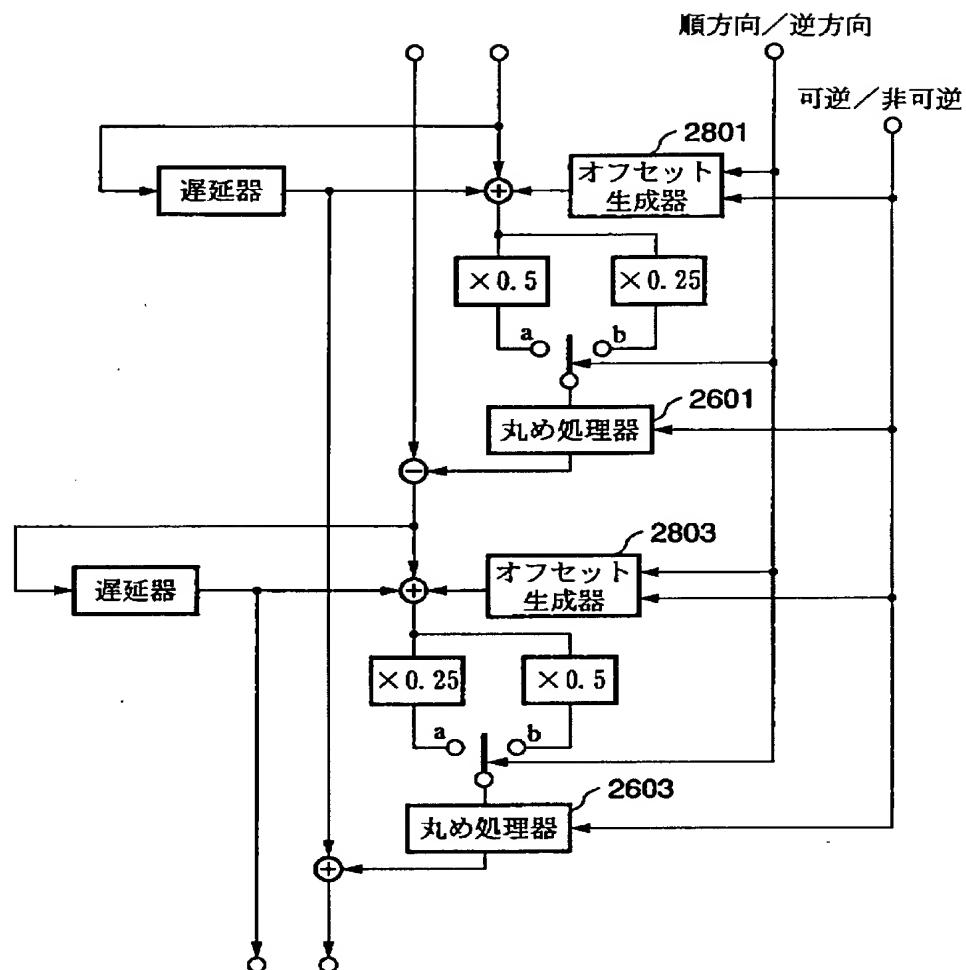
【図28】



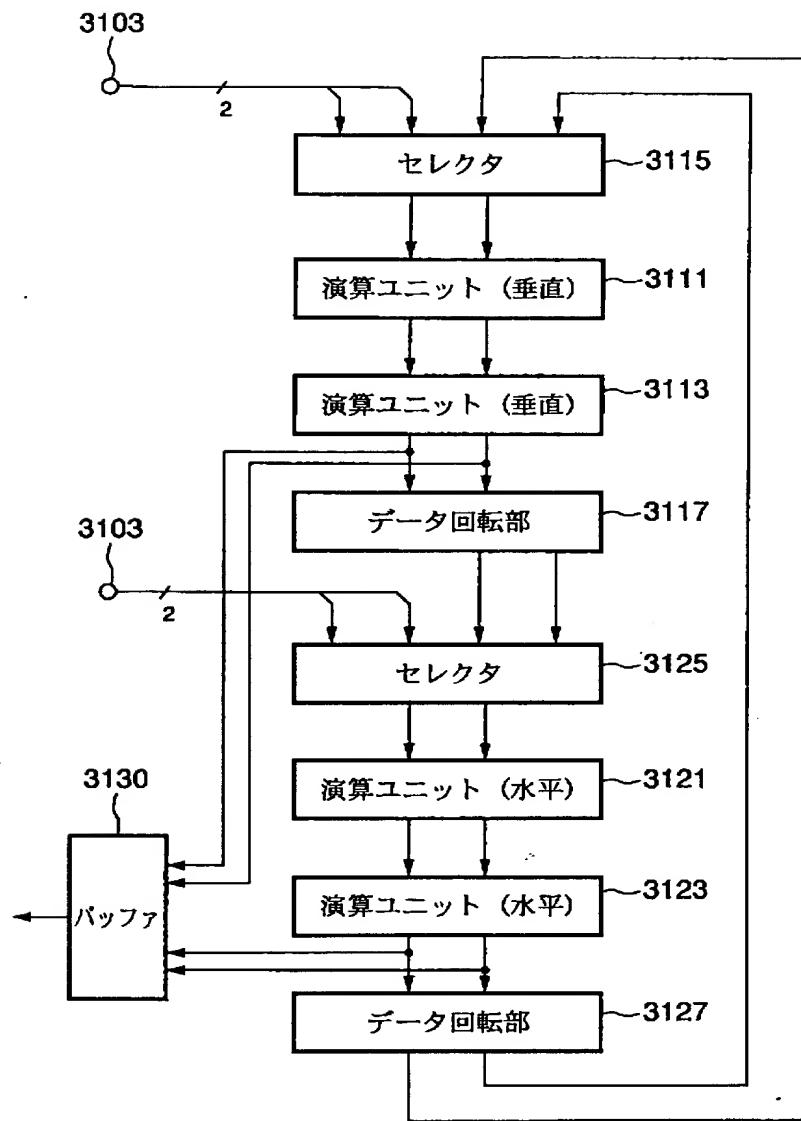
【図29】



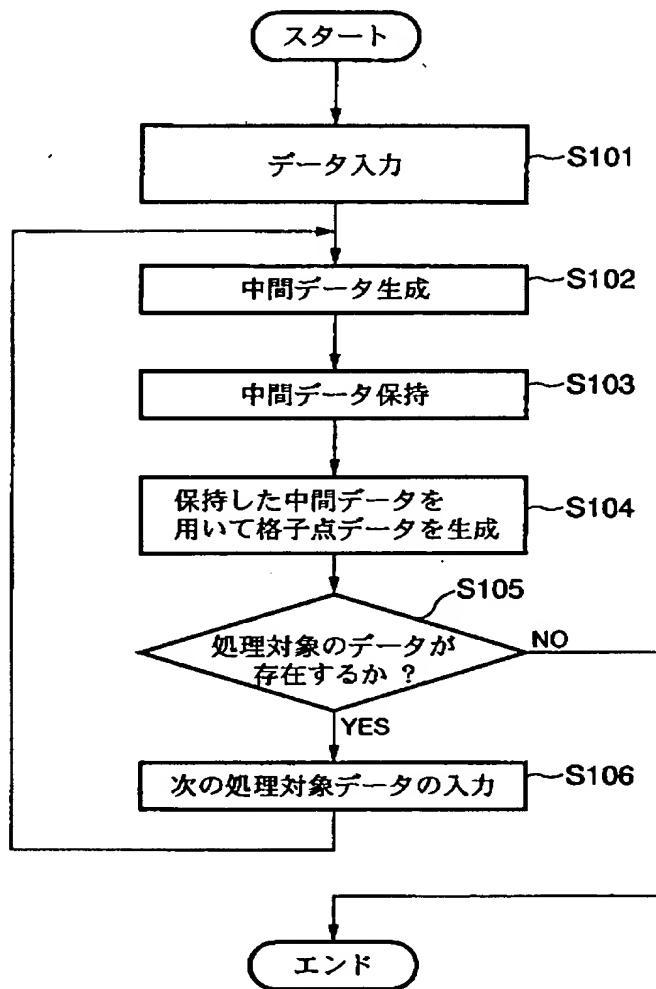
【図30】



【図31】



【図33】



【書類名】 要約書

【要約】

【課題】 効率的にウェーブレット変換処理を実行できるデータ処理装置及びその制御方法、データ処理方法、コンピュータ可読メモリを提供する。

【解決手段】 データを処理するデータ処理装置であって、

データを保持するレジスタ613と、複数の加算器615、617と、乗算器611を有する演算ユニット621とを備え、演算ユニット621は、該乗算器の乗算係数が異なる演算ユニット群622～624と従属に接続した構成からなり、該演算ユニット群内の前記各構成要素は、 $2^n + 1$ タップと $2^n - 1$ タップのデータを入力して、ウェーブレット変換係数の演算を行うことが可能になるように構成されている。

【選択図】 図6

出願人履歴情報

識別番号 [000001007]

1. 変更年月日 1990年 8月30日

[変更理由] 新規登録

住 所 東京都大田区下丸子3丁目30番2号
氏 名 キヤノン株式会社



Creation date: 12-13-2004
Indexing Officer: AVU - ANHTRAM VU
Team: OIPEBackFileIndexing
Dossier: 09982916

Legal Date: 09-14-2004

No.	Doccode	Number of pages
1	CTRS	5

Total number of pages: 5

Remarks:

Order of re-scan issued on